TESIS

PENGARUH EKSTRAKSI FITUR DAN STEMMER TERHADAP ALGORITMA SUPPORT VECTOR MACHINE PADA ANALISIS SENTIMEN BERBASIS LEKSIKON

Disusun oleh:

Nama : Luthfi Nurul Huda
NIM : 21.55.2157
Konsentrasi : Business Intelligence

PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2024

TESIS

PENGARUH EKSTRAKSI FITUR DAN STEMMER TERHADAP ALGORITMA SUPPORT VECTOR MACHINE PADA ANALISIS SENTIMEN BERBASIS LEKSIKON

THE EFFECT OF FEATURE EXTRACTION AND STEMMER ON SUPPORT VECTOR MACHINE ALGORITHM IN LEXICON-BASED SENTIMENT ANALYSIS

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister

Disusun oleh:

Nama : **Luthfi Nurul Huda**
NIM : **21.55.2157**
Konsentrasi : **Business Intelligence**

PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2024

HALAMAN PENGESAHAN

PENGARUH EKSTRAKSI FITUR DAN STEMMER TERHADAP ALGORITMA SUPPORT VECTOR MACHINE PADA ANALISIS SENTIMEN BERBASIS LEKSIKON

THE EFFECT OF FEATURE EXTRACTION AND STEMMER ON SUPPORT VECTOR MACHINE ALGORITHM IN LEXICON-BASED SENTIMENT ANALYSIS

Dipersiapkan dan Disusun oleh

Luthfi Nurul Huda

21.55.2157

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Kamis, 02 Mei 2024

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, Kamis 02 Mei 2024
Rektor

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

HALAMAN PERSETUJUAN

PENGARUH EKSTRAKSI FITUR DAN STEMMER TERHADAP ALGORITMA SUPPORT VECTOR MACHINE PADA ANALISIS SENTIMEN BERBASIS LEKSIKON

THE EFFECT OF FEATURE EXTRACTION AND STEMMER ON SUPPORT VECTOR MACHINE ALGORITHM IN LEXICON-BASED SENTIMENT ANALYSIS

Dipersiapkan dan Disusun oleh

Luthfi Nurul Huda

21.55.2157

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Kamis, 02 Mei 2024

Pembimbing Utama

Dr. Andi Sunyoto, M, Kom.
NIK. 190302052

Anggota Tim Penguji

Dhani A., S.Kom., M.Kom., Ph.D.
NIK. 190302197

M. Hanafi, S.Kom., M.Eng., Ph.D.
NIK. 190302024

Pembimbing Pendamping

Kusnawi, S.Kom., M.Eng.
NIK. 190302112

Dr. Andi Sunyoto, M, Kom.
NIK. 190302052

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, Kamis 02 Mei 2024
Direktur Program Pascasarjana

Dr. Kusrini, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Luthfi Nurul Huda
NIM : 21.55.2157
Konsentrasi : Business Intelligence

Menyatakan bahwa Tesis dengan judul berikut:
Pengaruh Ekstraksi Fitur Dan Stemmer Terhadap Algoritma Support Vector Machine Pada Analisis Sentimen Berbasis Leksikon

Dosen Pembimbing Utama : Dr. Andi Sunyoto, M.Kom.
Dosen Pembimbing Pendamping : Kusnawi, S.Kom., M.Eng.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, Kamis 02 Mei 2024
Yang Menyatakan,

Luthfi Nurul Huda

HALAMAN PERSEMBAHAN

Segala puji dan syukur kehadirat Allah SWT atas berkah, rahmat dan hidayah-Nya yang senantiasa dilimpahakan kepada kami, sehingga dapat merampungkan Tesis yang berjudul "**Pengaruh Ekstraksi Fitur Dan Stemmer Terhadap Algoritma Support Vector Machine Pada Analisis Sentimen Berbasis Leksikon**". Semoga dapat diterima sebagai salah satu amal kebaikan.

Selaras dengan harapan kami. Penelitian ini kami persembahkan kepada kedua orang tua tekasih kami. Dengan segala bentuk dukungan lahir dan batin, kami bisa menyelesaikan studi Program Magister (S2) pada Program Studi Pendidikan Jarak Jauh Magister Teknik Informatika (PJJ-MTI) Universitas Amikom Yogyakarta.

HALAMAN MOTTO

"Live to learn how to live and die in peace" (1)

"The truth is we might not win every fight. But then still, we need to fight every fight. With all courage, till the last breath we have" (2)

(LnH)

KATA PENGANTAR

Segala puji dan syukur kehadirat Allah SWT atas berkah, rahmat dan hidayah-Nya yang senantiasa dilimpahakan kepada penulis, sehingga bisa merampungkan Tesis yang berjudul "**Pengaruh Ekstraksi Fitur Dan Stemmer Terhadap Algoritma Support Vector Machine Pada Analisis Sentimen Berbasis Leksikon**" ini dengan baik, sebagai syarat untuk menyelesaiakan Program Magister (S2) pada Program Studi Pendidikan Jarak Jauh Magister Teknik Informatika (PJJ-MTI) Universitas Amikom Yogyakarta.

Dalam penyusunan Tesis ini banyak hambatan serta rintangan yang penulis hadapi namun pada akhirnya dapat dilalui berkat adanya dukungan dan bantuan dari berbagai pihak baik secara moral maupun spiritual. Untuk itu pada kesempatan ini penulis menyampaikan ucapan terimakasih kepada:.

1. Bapak Prof. Dr. M. Suyanto, M.M., selaku Rektor Universitas Amikom Yogyakarta.
2. Ibu Prof. Dr. Kusrini, M.Kom., selaku Direktur Program Pasca Sarjana Universitas Amikom Yogyakarta yang telah memberikan kesempatan dan izin untuk menempuh studi lanjut di Program Studi Pendidikan Jarak Jauh Magister Teknik Informatika (PJJ-MTI).
3. Bapak Dr. Andi Sunyoto, M.Kom., selaku pembimbing utama. Yang bersedia menyempatkan waktu di sela-sela kesibukan beliau, untuk memberikan dukungan, membimbing, mengoreksi dan mengarahkan penulis demi kesempurnaaan penulisan penelitian ini.

4. Bapak Kusnawi, S.Kom., M.Eng., selaku pembimbing pendamping yang disela sela kesibukannya dapat memberikan arahan, pendapat, dan memberi semangat agar penulis dapat menyelesaikan penelitian.
5. Tim Penguji dari SPT, SHPT, hingga UT yang telah memberikan arahan dan wawasan lebih dalam proses penyempurnaan penulisan.
6. Seluruh Dosen Pengajar di Program Studi Pendidikan Jarak Jauh Magister Teknik Informatika (PJJ-MTI) Universitas Amikom Yogyakarta dari semester pertama hingga terakhir yang memberikan arahan, dukungan, semangat, dan sharing pengetahuan sehingga penulis mendapatkan wawasan baru yang lebih luas dalam menyelesaikan tugas disetiap studi.
7. Segenap Civitas Akademika (Pengelola dan Admisi) Program Studi Pendidikan Jarak Jauh Magister Teknik Informatika (PJJ-MTI) Universitas Amikom Yogyakarta yang telah memberikan pelayanan dan bantuan sangat baik dalam kebutuhan studi.
8. Ibu dan Bapak terkasih, terimakasih atas segala doa dan dukungan. Penulis percaya dalam setiap kemudahan dan kesuksesan penulis bukanlah karena jerih payah penulis, melaikan berkat secercah do'a kedua orang tua yang menembus langit.
9. Teman-teman terdekat, rekan-rekan dan guru-guru penulis utamanya yang berada dibawah naungan Pondok Pesantren Nurul jadid, terkhusus Fakultas Teknik Universitas Nurul Jadid. Terimakasih atas kesediannya berbagi keluh kesah, sedih seneng, tawa duka. Penulis tidak akan sampai pada titik ini tanpa dukungan dan dampingan kalian selama ini. Salam hormat penulis.

10. Teman-teman Mahasiswa Program Studi Pendidikan Jarak Jauh Magister Teknik Informatika (PJJ-MTI) Universitas Amikom Yogyakarta Angkatan 2021 Genap yang telah memberikan pengalaman, suasana dan keluarga baru.
11. Last but not least, I wanna thank me. I wanna thank me for believin' in me. I wanna thank me for doing all this hard work. I wanna thank me for having no days off. I wanna thank me for, for never quitting. I wanna thank me for always bein' a giver. And tryna give more than I receive. I wanna thank me for tryna do more right than wrong. I wanna thank me for just bein' me at all times. By Snoop Dogg.

Dengan seneng hati penulis menerima kritik dan saran yang membangun dari pembaca. Karena penulis menyadari dengan penuh bahwa dalam penyusunan penelitian ini masih banyak kekurangan. Akhir kata, semoga penelitian ini dapat memberikan manfaat bagi pembacanya.

Yogyakarta, 05 Mei 2024

Penulis

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

INTISARI

Analisis sentimen adalah bidang yang memiliki potensi besar dalam penelitian dan aplikasi praktis. Berbagai tantangan dihadapi dalam analisis sentimen, termasuk bagaimana menentukan kombinasi preprocessing yang optimal, menentukan algoritma klasifikasi terbaik, bagaimana meningkatkan akurasi algoritma yang digunakan, dan bahkan bagaimana data dibersihkan. Penelitian ini sendiri bertujuan untuk mencari kombinasi terbaik dari ekstraksi fitur stemmer dalam analisis sentimen berbasis leksikon terhadap algoritma SVM. Terdapat 6 skenario yang digunakan dalam penelitian ini antara kombinasi stemmer dan ekstraksi fitur. Stemmer yang digunakan dalam penelitian ini adalah *Porter stemmer*, *Snowball Stemmer* dan *Wordnet Lemmatizer*. *N-Grams* dan *BoW* dipilih sebagai ekstraksi fitur yang akan diuji. Berdasarkan tujuan dari penelitian ini, *SentiWordNet* dipilih sebagai kamus leksikon dan SVM sebagai algoritma klasifikasi yang dipilih. Untuk memperkuat hasil penelitian, semua skenario diuji pada 3 dataset yang berbeda yaitu dataset *IMDb*, dataset *Twitter US Airline Sentiment* dan dataset *Stanford140*. Metode yang diusulkan dalam penelitian ini dinilai efektif dalam mengklasifikasikan teks sentimen. Hal ini dibuktikan dengan akurasi pada dataset IMDb dengan perpaduan *Lemmatizer* dan *N-Grams*, *Twitter US Airline Sentiment*, dan *Sentiment140* dengan perpaduan *Porter* dan *N-Grams*. Masing-masing mencatat akurasi sebesar 96.08%, 88.46% dan 91.71%. Dan *Porter stemmer* dan *N-Gram* merupakan kombinasi *stemmer* dan *feature extraction* yang paling efektif meningkatkan performa SVM, dengan rata-rata akurasi 91.35% pada ketiga dataset yang digunakan.

Kata kunci: analisis sentimen, *stemmer*, *feature extraction*, *lexicon*, SVM

ABSTRACT

Sentiment analysis is a field that has great potential in research and practical applications. Various challenges are faced in sentiment analysis, including how to determine the optimal combination of preprocessing, determine the best classification algorithm, how to improve the accuracy of the algorithm used, and even how the data is cleaned. This research itself aims to find the best combination of stemmer feature extraction in lexicon-based sentiment analysis against SVM algorithm. There are 6 scenarios used in this research between the combination of stemmer and feature extraction. The stemmers used in this research are Porter stemmer, Snowball Stemmer and Wordnet Lemmatizer. N-Gram and BoW were chosen as the feature extraction to be tested. Based on the purpose of this research, SentiWordNet was chosen as the dictionary lexicon and SVM as the selected ML algorithm. To strengthen the results of the study, all scenarios were tested on 3 different datasets namely IMDb dataset, Twitter US Airline Sentiment dataset and Stanford140 dataset. The method proposed in this research is considered effective in classifying sentiment text. It is proven by the accuracy on IMDb dataset with Lemmatizer and N-Gram blend, Twitter US Airline Sentiment, and Sentiment140 with Porter and N-Gram blend. Each recorded an accuracy of 96.08%, 88.46% and 91.71%. And Porter stemmer and N-Gram is a combination of stemmer and feature extraction that most effectively improves SVM performance, with an average accuracy of 91.35% on the three datasets used.

Keyword: sentiment analysis, stemmer, feature extraction, lexicon, SVM

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Analisis sentimen merupakan sebuah bidang ilmu yang memiliki potensi besar dalam penelitian dan aplikasi praktis. Membantu menjelaskan bahwa penelitian analisis sentimen khusunya teks semakin menarik perhatian baik di dalam maupun di luar negeri (Han et al., 2020). Ini merupakan tugas penting untuk mendeteksi polaritas sentimen dalam bentuk teks, yang banyak diterapkan dalam sistem e-*commerce*, blog, dan sosial media (Nurcahyawati & Mustaffa, 2023). Secara umum analisis sentimen merupakan proses atau cara untuk mengevaluasi sejauh mana bahasa tertulis atau secara lisan ditentukan sebagai ekspresi positif, negatif, maumpun netral (Tamara & Milićević, 2018). 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 (Zou et al., 2016). Teknik ini dapat dibagi menjadi metode berbasis aturan (*rule-based methods*) dan berbasis statistik (*statistical-based methods*) (Chen et al., 2019). Saat ini, *machine-learning* merupakan metode utama yang diterapkan pada analisis sentimen. Sebagai acuan apakah analisis sentimen mendapatkan hasil yang baik. Penelitian terkait analisis sentimen sudah banyak dilakukan, untuk mendapatkan kesimpulan berdasarkan data yang diolah atau didapatkan dari berbagai sumber *data text*.

Beberapa tahapan menjadi tantangan dalam melakukan analisis sentimen. Baik pada *preprocessing* data, pemilihan leksikon, *feature extraction* dan lain sebagainya. Resyanto et al., (2019) menyatakan *stemming* mampu menghasilkan akurasi tertinggi. Di lain sisi teknik atau algoritma klasifikasi berbasis leksikon (*lexicon based*) efektif untuk meningkatkan akurasi lebih lanjut (Rani et al., 2021). Terakhir Fikri & Sarno, (2019) menyatakan gabungan antara *feature extraction* dan algoritma klasifikiasi SVM dapat meningkatkan baik *accuracy* dan F1-*score*.

Saat ini, banyak penelitian tentang analisis sentimen telah dikerjakan. Berbagai tantangan dihadapi, antara lain bagaimana menentukan kombinasi antara *preprocessing* yang optimal, menentukan algoritma klasifikasi terbaik, bagaimana meningkatkan akurasi dari algoritma yang digunakan, dan bahkan bagaimana data dibersihkan. Seperti penelitian-penelitian yang akan dipaparkan berikut.

Ahuja et al., (2019) menyimpulkan dalam penelitiannya bahwasannya *feature extraction* yakni TF-IDF dan N-Gram memberikan output maksimal pada empat parameter perbandingan yakni *accuracy*, *precision*, *recall*, dan F1-*score*. Khususnya pada *feature extraction* TF-IDF menaikkan kinerja analisis sentimen sebanyak 3-4% lebih tinggi dari pada menggunakan N-*gram* terhadap algoritma yang digunakan. Penggunaan *Bag-of-Word* (BOW) pada analisis sentimen menggunakan NBC oleh Gowri et al., (2022) mendapatkan hasil akhir yang akurat, dan prosesnya juga relatif cepat. Dan memprediksi akurasi terbaik berdasarkan hasil ulasan film menggunakan *Naïve Bayes Classifier*. *Part-Of-Speech Tagging* (POST) dan *Term-Frequency* (TF) bekerja dengan baik pada algortima *Support Vector Machine* (SVM) dengan menunjukkan akurasi sebesar 80% (Barve et al., 2022).

Hasil eksperimen dari Kumar & Subba, (2020) dengan kerangka kerja menggunakan *TfidfVectorizer* dan pengklasifikasi SVM pada dataset *IMDB movie review* and *Amazon electronic items review* mencapai kinerja tinggi dalam mengkasifikasikan sentimen. Resyanto et al., (2019) melakukan ekperimen dengan membandingkan akurasi yang dihasilkan dari berbagai tahap *preprocessing* pada analisis sentimen menggunakan *Naïve Bayes*. Hasil yang ditunjukkan adalah prosess *stemming* mendapatkan akurasi paling tinggi diantara semua tahap *preprocessing* yakni sebesar 70%. Terakhir penelitian oleh Nurcahyawati & Mustaffa, (2023) melakukan eksperimen terhadap analisis sentimen menggunakan *feature extraction* TF-IDF, PCA, FM dan EDS pada algoritma SVM. Menyatakan bahwa gabungan antara SVM dan FM mendapatkan akurasi paling tinggi dengan nilai 96%, yang mana juga dipengaruhi oleh *preprocessing* khususnya tahap *stimming*.

Terdapat berbagai algoritma klasifikasi berbeda diantaranya, *machine learning*, *lexicon based*, *ensemble classifier* dan *neural network-based classifier* yang digunakan oleh para peneliti. Teknik atau algoritma klasifikasi berbasis leksikon (*lexicon based*) efektif untuk meningkatkan akurasi lebih lanjut (Rani et al., 2021). Leksikon digabungn sebagai penentu skor sentimen pada teks (Ayu et al., 2019). Pada dasarnya teknik klasifikasi berbasis leksikon atau kamus, menghitung teks dengan menggunakan kamus sentimen. Dalam pendekatan ini, skor dari polaritas setiap kata BOG dihitung dengan menggunakan kamus polaritas yang sudah ada seperti *SentiWordNet* dan *WordNet* (Rani et al., 2021). Dengan

adanya leksikon atau kamus ini peneliti tidak perlu melakukan pelabelan secara manual pada setiap kata yang akan diproses untuk analisis sentimen.

Sebelum proses analisa sentimen dilakukan, data mentah sebelumnya akan melewati proses *preprocessing*. Peranan *preprocessing* sangat penting dalam proses data mining khususnya analisis sentimen. Mengacu pada pembersihan (*cleaning*) data dari informasi tidak berguna dan tidak akan membantu dalam proses *training* dan dapat menyebabkan kebingungan (*confusion*) dalam proses klasifikasi (Qaisar, 2020). Menurut Sethi et al., (2020) *preprocessing* harus dilakukan terlebih dahulu sebelum benar-benar dilakukan proses *feature extraction*. Karena data mentah mengandung banyak *noise*, kata-kata salah eja, dan menyertakan berbagai singkatan serta kata-kata *slang*. Ini sering kali mengganggu sentimen dan dapat menurunkan performa pengklasifikasian.

Preprocessing berisi rangkaian prosedur untuk membersihkan data. Serangkaian teknik yang merupakan teknik dasar dalam *preprocessing* diantaranya *tokenizing*, *stop words removal* atau *elimination*, *transform case* dan *stemming* (Nurcahyawati & Mustaffa, 2023). Sedangkan Rustam et al., (2019) memaparkan *preprocessing* dengan teknik yang lebih kompleks. Berisi beberapa teknik yaknni *punctuation removal*, *numerical removal*, *convert to lower case*, *stemming* serta *stopwords removal*. Pada dua penelitian sebelumnya yang terpapar menyatakan bahwa *stemming* meningkatkan kinerja dari algoritma *classifier* yang digunakan dan dapat meningkatkan akurasi.

Setelah data melewati *preprocessing*, proses berikutnya adalah *feature extraction*. Proses ini digunakan untuk mengompres data secara lebih *compact* dari

data mentah sehingga redundansi dihilangkan sembari mempertahankan informasi yang relevan. Pola data akan lebih mudah ditemukan sehingga memudahkan tugas klasifikasi. *Feature extraction* yang baik diperlukan untuk memiliki sifat diskriminatif, untuk memaksimalkan variabilitas antara kelas dan meminimalkan variablitas intra kelas (Ferdiana et al., 2019). Secara sederhana Gowri et al., (2022) menjelaskan *feature extraction* merupakan pendekatan di mana data disiapkan untuk data tabular. Semua transformasi data dijalankan secara pararel dengan inputan data mentah dan kemudian digabung untuk membentuk suatu set data besar. Proses berikutnya merupakan *classification* dan *evaluation*, yang pada penelitian ini menggunakan algoritma *Support Vector Machine* (SVM).

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 sedarhana 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-*linear* diubah menjadi titik yang dapat dipisahkan secara *linear* (Obiedat et al., 2022). SVM sudah terbukti sebagai salah satu algoritma *supervised* paling kuat untuk pengkategorisasian teks (Rahat et al., 2019). Dan dalam kasus ini *performance* digunakan sebagai alat ukur kinerja dari SVM.

Berdasarkan latar belakang yang menjelaskan tentang beberapa tantangan dalam analisis sentimen penelitian ini akan mengangkat tentang bagaimana

meningkatkan algoritma yang digunakan dalam hal ini algoritma *Support Vector Machine* (SVM). Berdasarkan penelitian-penelitian sebelumnya seperti penelitian oleh Ahuja et al., (2019) menjelaskan pengaruh *feature extraction* yang memberikan output maksimal pada algoritma yang digunakan. Serta penelitian yang dilakukan oleh Nurcahyawati & Mustaffa, (2023), Rustam et al., (2019), dan Resyanto et al., (2019) menyatakan bahwa *stemming* meningkatkan kinerja dari algoritma *classifier* yang digunakan dan dapat meningkatkan akurasi.

Penelitian-penelitian tersebut menunjukkan bahwa *feature extraction* dan *stemming* dapat meningkatkan kinerja SVM dalam analisis sentimen. Namun, belum ada penelitian yang secara sistematis mengevaluasi pengaruh kombinasi *feature extraction* dan *stemmer* (jenis *stemming*) terhadap kinerja SVM. Maka penelitian ini akan mengkombinasikan antara *feature extraction* dan *stemmer* untuk mengukur pengaruh dari kombinasi keduanya terhadap kinerja algoritma *Support Vector Machine* pada analisis sentimen berbasis leksikon. Kemudian hasil akhir penelitian ini akan menunjukkan *accuracy, precision, recall* dan *F1-Score* dari algoritma SVM dengan menggunakan beberapa skenario gabungan kombinasi antar *stemmer* dan *feature extraction*.

1.2. Rumusan Masalah

Berdasarkan latar belakang terpapar dibagian 1, maka rumusan masalah pada penelitian ini adalah sebagai berikut.

a. Apakah *stemmer* dan *feature extraction* berpengaruh terhadap nilai *accuracy, precision, recall* dan *F1-Score* pada algoritma *Support Vector Machine* (SVM) dalam analisis sentimen berbasis leksikon?
b. Metode *stemmer* dan *feature extraction* manakah yang paling efektif dalam meningkatkan kinerja algoritma SVM?

1.3. Batasan Masalah

Berikut batasan-batasan masalah pada penelitian ini:

a. Proses pengolahan dan klasifikasi data menggunakan bahasa *python* pada platform *Visual Studio Code*
b. Dataset yang digunakan adalah IMDB's Movie Review, Twitter US Airline Sentiment, Stanford Sentiment140
c. Algoritma klasifikasi yang digunakan *Support Vector Machine*
d. *Stemmer* yang digunakan ialah *Porter Stemmer*, *Snowball Stemmer*, dan *WordNet Lemmatizer*
e. *Feature extraction* yang digunakan adalah *N-grams*, dan *Bag-of-Words* (BoW)
f. *Lexicon* yang digunakan ialah *SentiWordNet*

1.4. Tujuan Penelitian

Bagian ini memuat penjelasan secara spesifik:

a. Mengukur pengaruh *stemmer* dan *feature extraction* terhadap nilai *accuracy, precision, recall* dan *F1-Score* pada algoritma *Support Vector Machine* dalam analisis sentimen berbasis leksikon

b. Menentukan metode *stemmer* dan *feature extraction* yang optimal untuk algoritma *Support Vector Machine* dalam analisis sentimen berbasis leksikon

1.5. Manfaat Penelitian

Manfaat yang dapat diambil pada penelitian ini adalah sebagai berikut:

a. Membantu meningkatkan kinerja algoritma *Support Vector Machine* dalam analisis sentimen berbasis leksikon
b. Membantu mengidentifikasi metode *stemmer* dan *feature extraction* yang optimal untuk algoritma *Support Vector Machine* dalam analisis sentimen berbasis leksikon
c. Membantu mengidentifikasi faktor-faktor yang mempengaruhi kinerja algoritma *Support Vector Machine* dalam analisis sentimen berbasis leksikon
d. Dapat menjadi referensi penggunaan *stemmer* dan *feature extraction* pada analisis sentimen dengan algoritma *Support Vector Machine*
e. Dapat menjadi rujukan terhadap penelitian berikutnya yang berkaitan tentang analisis sentimen

1.6. Hipotesis

Berikut adalah hipotesis dari penelitian ini, yakni:

- H1: *Stemming* akan meningkatkan kinerja algoritma SVM dalam sentimen berbasis leksikon.
- H2: *Feature extraction* bisa meningkatkan kinerja algoritma SVM dalam analisis sentimen berbasis leksikon.
- H3: Gabungan antara *Stemming* dan *Feature extraction* yang tepat dapat meningkatkan kinerja algoritma SVM dalam analisis sentimen berbasis leksikon

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Penelitian yang dilakukan mengacu pada penelitian-penelitian sebelumnya sebagai tinjauan pustaka untuk mendukung dan sebagai penyempurna dari penelitian-penelitian sebelumnya. Berikut adalah beberapa penelitian-penelitian yang menjadi referensi bagi penelitian yang dilakukan.

Ahuja et al., (2019) pada penelitiannya membahas dampak dari *feature extraction* yang berbeda yakni TF-IDF dan BOW pada kinerja analisis sentimen. Menerapkan 6 tahap preprocessing sebelum melakukan *feature extraction* yakni *Tokenization, Normalization, Stemming, Lemmatization, Removing Stop Words* dan *Noise removal*. Hasil dari penelitian ini menunjukkan bahwasannya *feature extraction* yakni TF-IDF dan N-*Gram* memberikan output maksimal pada empat parameter perbandingan yakni *accuracy*, *precision*, *recall*, dan F1-*score*. Khususnya pada *feature extraction* TF-IDF menaikkan kinerja analisis sentimen sebanyak 3-4% lebih tinggi dari pada menggunakan N-*gram* terhadap algoritma yang digunakan. Saran dari penelitian ini adalah melakukan perbandingan fitur lain seperti fitur skor polaritas kata, penyematan kata, fitur khusus twitter, dll.

Penelitian berikutnyan bertujuan untuk memperkirakan keseluruhan emosi sebuah ulasan berdasarkan hubungan antar kata di dalam analisis sentimen ke serangkaian ulasan film yang ditulis manusia untuk menentukan apakah reaksi keseluruhan mereka terhadap film itu positif, negatif, atau netral. Penelitian ini

berfokus kepada penggunaan BoW dan algoritma *Naïve Bayes* untuk meningkatkan hasil dari analisis sentimen. Hasil dari penelitian ini menunjukkan, gabungan antara algoritma *Naïve Bayes* dan teknik BoW mengatasi masalah probabilitas nol dengan mengasumsikan bahwa setiap kata dalam data uji diulang setidaknya sekali dan berhasil mencapai akurasi sebesar 72%. Saran pada penelitian ini adalah memperluas penelitian untuk mencakup komentar sarkasme, slang berbeda dalam bahasa yang sama, dan kata-kata yang salah eja (Gowri et al., 2022).

Selanjutnya Barve et al., (2022) menggunakan pembelajaran inkremental untuk merekam perubahan sentimental yang terjadi pada data dengan menghasilkan "*Bag-of-Words*" sentimental dalam domain kesehatan dan dengan demikian mendeteksi perubahan persentase informasi yang salah pada interval waktu yang berbeda. Fokus penelitian ini adalah pada *feature extraction* BOW (*Pos tagging* dan TF) pada analisis sentimen domain Kesehatan. Penelitian ini menunjukkan bahwasanya model *POS Tanging* dan *TF* bekerja dengan baik dengan mesin SVM menunjukkan akurasi 80% sedangkan pada Decision Tree sebesar 56.66%.

Kumar & Subba, (2020) mengusulkan *framework* analisis sentimen berbasis *TfidfVectorize* dan SVM untuk menganalisis sentimen dokumen dalam data teks. Penelitian ini menggunakan dataset dari *Amazon's electronic item review IMDB*. Pada dataset yang digunakan *framework* yang diusung (*TfidfVectorizer* dan SVM) pada analisis sentimen mencapai kinerja tinggi dengan akurasi sebesar 0.915. Saran dari penelitian ini adalah untuk berfokus pada proses pengoptimalan agar mengurangi kompleksitas komputasi kerangka kerja yang diusulkan dengan tingkat rentang ngram yang lebih tinggi.

Penelitian berikutnya berfokus pada perbandingan hasil akurasi pada setiap preprocessing data menggunakan *Naïve Bayes* untuk analisis sentimen. Mengusulkan kombinasi metode *preprocessing* untuk analisis sentimen pada *Twitter review iPhone*. Hasil penelitian menunjukkan bahwa kombinasi dari 5 metode: *URL removal, emoticon handling, case folding, expressive lengthening and stemming,* merupakan metode paling optimal dengan akurasi 70.88%. Kekurangan dari penelitian ini hanya menggunakan satu algoritma klasifikasi yakni *Naïve Bayes* (Resyanto et al., 2019).

Penelitian terakhir oleh Nurcahyawati & Mustaffa, (2023) melakukan eksperimen untuk menentukan kombinasi antara preprocessing, algoritma pada analisis sentimen. Tujuan dari penelitian ini untuk menentukan kombinasi teknik *preprocessing* yang optimal, bagaimana membersihkan dataset, dan menentukan algoritma klasifikasi terbaik. Hasil dari penelitian ini menunjukkan bahwa kombinasi SVM-FM menghasilkan akurasi terbaik sebesar 96%. Dan dengan kombinasi teknik *preprocessing tokenizing, stop words elimination, transform case,* and *stemming*. Saran pada penelitian ini ialah untuk menggunakan *feature extraction* yang lebih variatif.

Berdasarkan penelitian-penelitian sebelumnya, perbedaan penelitian yang akan dilakukan dengan penelitian-penelitian sebelumnya ialah penelitian ini berfokus pada implementai *stemmer* dan *feature extraction* terhadap performa algoritma SVM pada analisis sentiment berbasis leksikon. Dan hasil akhir penelitian ini akan menunjukkan *accuracy, precision, recall* dan *F1-Score* dari

algoritma SVM dengan menggunakan beberapa proses *stemmer* dan *feature extraction*.

2.2. Keaslian Penelitian

Tabel 2.1. Matriks literatur review dan posisi penelitian
Pengaruh Ekstraksi Fitur Dan Stemmer Terhadap Algoritma Support Vector Machine Pada Analisis Sentimen Berbasis Leksikon

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	The impact of features extraction on the sentiment analysis	(Ahuja et al., 2019) Procedia Computer Science	Membahas dampak fitur yang berbeda (TF-IDF dan Vektor N-GRAM) pada kinerja analisis sentimen.	Membandingkan kinerja berdasarkan Precission, Recall, Accuracy, dan F-Score. Dengan hasil 3-4% lebih besar.	Perbandingan fitur lain seperti fitur skor polaritas kata, penyematan kata, fitur khusus twitter, dll.	Penelitian ini membandingkan metode KNN, Decision Tree, SVM, Logistic Regression, Naïve Bayes, Random Forest dengan dampak fitur Pembobotan TF-IDF dan Vektor N-GRAM. Sedangkan pada penelitian yang diajukan akan berfokus pada satu metode klasifikasi yakni SVM dengan dampak stemming dan feature extraction.
2	Improved Sentimental Analysis to the Movie Reviews using Naive Bayes Classifier	(Gowri et al., 2022) Proceedings of the International Conference on Electronics and Renewable Systems	Memperkirakan keseluruhan emosi sebuah ulasan berdasarkan hubungan antar kata di dalam analisis sentimen ke serangkaian ulasan film yang ditulis manusia untuk menentukan apakah reaksi keseluruhan mereka terhadap film itu positif, negatif, atau netral	Gabungan antara algoritma NB dan teknik BoW mengatasi masalah probabilitas nol dengan mengasumsikan bahwa setiap kata dalam data uji diulang setidaknya sekali.	Diperluas untuk mencakup komentar sarkasme, slang berbeda dalam bahasa yang sama, dan kata-kata yang salah eja.	Penelitian ini berfokus kepada penggunaan BoW dan algoritma Naïve Bayes pada analisis sentimen. Sedangkan penelitian yang diajukan akan berfokus pada algoritma SVM dengan dapat dari stemminf dan feature extraction pada performanya.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
3	A Novel Evolving Sentimental Bag-of-Words Approach for Feature Extraction to Detect Misinformation	(Barve et al., 2022) International Journal of Advanced Computer Science and Applications	Menggunakan pembelajaran inkremental untuk merekam perubahan sentimental yang terjadi pada data dengan menghasilkan "Bag-of-Words" sentimental dalam domain kesehatan dan dengan demikian mendeteksi perubahan persentase informasi yang salah pada interval waktu yang berbeda.	Model POS Tanging dan TF bekerja dengan baik dengan mesin SVM menunjukkan akurasi 80% sedangkan pada Decision Tree sebesar 56.66%.	Pendekatan n-gram dapat digunakan untuk memprediksi urutan kemunculan kata dengan benar, ukuran dataset relatif kecil.	Fokus penelitian ini adalah pada feature extraction BOW (Pos tagging dan TF) pada analisis sentimen domain Kesehatan. Sedangkan penelitian yang diajukan akan berfokus kepada penggunaan stemmer dan feature extraction untuk performa algoritma SVM pada analisis sentimen berbasis lexicon.
4	A tfidfvectorizer and SVM based sentiment analysis framework for text data corpus	(Kumar & Subba, 2020) 26th National Conference on Communications	Mengusulkan framework analisis sentimen berbasis TfidfVectorize dan SVM baru untuk menganalisis sentimen dokumen dalam data teks.	Pada dataset ulasan elektronik Amazon dan korpus ulasan fil IMDB menunjukkan framework yang diusung (TfidfVectorizer dan SVM) pada analisis sentimen mencapai kinerja tinggi.	Berfokus pada proses pengoptimalan untuk mengurangi kompleksitas komputasi kerangka kerja yang diusulkan dengan tingkat rentang ngram yang lebih tinggi	Penelitian ini mengajukan framework (TfidfVectorizer dan SVM) pada analisis sentimen. Sedangkan penelitian yang akan dilakukan mengajukan stemmer dan feature extraction pada SVM untuk analisis sentimen berbasis lexicon.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
5	Choosing The Most Optimum Text Preprocessing Method for Sentiment Analysis: Case:iPhone Tweets.	(Resyanto et al., 2019) Proceedings of 2019 4th International Conference on Informatics and Computing	Mengusulkan kombinasi metode preprocessing untuk analisis sentimen pada twitter review iPhone.	Hasil penelitian menunjukkan bahwa kombinasi dari 5 metode : URL removal, emoticon handling, case folding, expressive lengthening and stemming, merupakan metode paling optimal dengan akurasi 70.88%.	Hanya menggunakan satu algoritma klasifikasi yakni Naïve Bayes.	Penelitian ini berfokus pada perbandingan hasil akurasi pada setiap preprocessing data menggunakan naïve bayes untuk analisis sentimen. Sedangkan pada penelitian yang diajukan akan berfokus pada penggunaan feature extraction dan stemmer untuk mencari performa paling tinggi pada klasifikasi sentimen dengan SVM.
6	Improving sentiment reviews classification performance using support vector machine-fuzzy matching algorithm.	(Nurcahyawati & Mustaffa, 2023) Bulletin of Electrical Engineering and Informatics	menentukan kombinasi teknik preprocessing yang optimal, bagaimana membersihkan dataset, dan menentukan algoritma klasifikasi terbaik.	Hasil dari penelitian ini menunjukkan bahwa kombinasi SVM-FM menghasilkan akurasi terbaik sebesar 96%. Dan dengan kombinasi teknik preprocessing okenizing, stop words elimination, transform case, and stemming.	Menggunakan feature extraction yang lebih variatif.	Penelitian ini melakukan eksperimen untuk menentukan kombinasi antara preprocessing, algoritma pada analisis sentimen. Sedangkan penelitian yang akan dilakukan berfokus mencari performa terbaik pada SVM dengan kombinasi stemmer dan feature extraction pada analisis sentimen berbasis lexicon.

2.3. Landasan Teori

2.3.1. Analisis Sentimen

Analisis sentimen merupakan salah satu bidang penelitian yang paling banyak dipelajari yang menggabungkan *Natural Language Processing* (NLP), *data mining* dan *web mining*. Penelitian tentang analisis sentimen sudah menyebar ke berbagai aspek seperti manajemen dan sosial karena dianggap penting untuk sebuah bisnis dan masyarakat (Obiedat et al., 2022). Analisis sentimen sendiri merupkan 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 algoritme untuk mengekstrak konten teks (Dangi et al., 2022).

Analisis sentimen dapat dibagi menjadi tiga teknik: teknik berbasis leksikon (*lexicon-based*), teknik berbasis machine learning (*machine learning-based*), dan teknik *hybrid-based* (Muhammad et al., 2021).

a) *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) (Shofiya & Abidi, 2021).

b) *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* (Shofiya & Abidi, 2021).

c) *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, efesiensi, dan skalabilitas (Shofiya & Abidi, 2021). Cara kerja dari pendekatan ini adalah menggunakan leksikon untuk mengidentifikasi kata-kata yang memeiliki kecenderungan tertentu dan kemudian menggunakan pendekatan *machine learning* untuk mempelajari hubungan antara kata-kata dengan sentimen teks.

2.3.2. Emosi

Emosi adalah perasaan dan pikirian subjektif kita (manusia). Emosi dipelajari di berbagai bidang misalnya psikologi, filsafat dan susiologi. Studinya sangat luas, mulai dari rekasi fisiologis (perubahan detak jantung, tekanan darah dan sebagainya), ekspresi wajah, gerak tubuh dan postur sehingga berbagai jenis pengalaman subjektif dari keadaan pikiran seseorang. Manusia sendiri memiliki enam emosi primer, yakni cinta, kegembiraan, keterkejutan, kemarahan, kesedihan

dan ketakutan yang dapat dibagi menjadi banyak emosi skunder dan tersier (Liu, 2012).

Emosi sendiri berkaitan erat dengan sentimen. Kekuatan sentimen atau pendapat biasanya terkatit dengan intensitas emosi tertentu misalnya kegembiraan (*joy*) dan kemarahan (*anger*). Pendapat yang kita pelajari dalam analisis sentimen kebanyakan adalah evaluasi (walaupun tidak selalu) (Liu, 2012). Secara umum kita bisa membagi emosi menjadi dua yakni emosi positif dan emosi negatif.

a) Emosi positif merupakan emosi yang mendatangkan perasaan positif. Seperti bahagia, cinta, harapan dan lain sebagainya.
b) Emosi negatif berbanding terbalik dengan emosi positif, emosi negatif identik dengan perasaan yang tidak menyenangkan. Seperti takut, kecewa, sedih, gelisah dan lain sebagainya.

2.3.3. 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 (Resyanto et al., 2019). Handayani et al., (2020) mengatakan dalam penelitiannya bahwa *preprocessing* merupakan awal yang sangat penting dalam pengumpulan data dari *text mining* digunakan untuk mengubah data mengikuti format yang dibutuhkan. Proses tersebut 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.

Sebelum data dimasukkan harus diproses terlebih dahulu dengan menerapkan tahapan *preprocessing* yang disepakati terlebih dahulu sebelum melanjutkan pada proses *text mining* selanjutnya. Tahapan *preprocessing* meliputi:

a) *Case Folding*

Tahapan *case folding* berfungsi *untuk* mengganti semua *letters* (huruf) yang ada pada kalimat menjadi huruf kecil (Resyanto et al., 2019)

b) *Cleaning*

Cleaning adalah tahapan untuk pembersihan kata dengan menghilangkan tanda baca seperti titik (.), koma (,) dan yang lain dengan tujuan untuk mengurangi noise (Handayani et al., 2020)

c) *Tokenizing*

Tahapan ini memiliki fungsi utama untuk membagi teks menjadi *token* (kata) terpisah untuk menetapkan indeks unik pada setiap kata (AlBadani et al., 2022).

d) *Stopword Removal*

Tahapan ini digunakan untuk menghilangkan kata yang tidak sesuai dengan topik dokumen, karena kata tersebut tidak mempengaruhi akurasi dalam klasifikasi sentimen (Fitriyyah et al., 2019). Mengatakan *stopwords* untuk menghilangkan konjungsi, kata ganti orang, dan kata lain yang tidak memiliki arti penting(Resyanto et al., 2019)

e) *Stemming*

Stemming merupakan proses yang digunakan untuk mengidentifikasi kata-kata dengan arti yang serupa. Ini akan membantu mengurangi redudansi untuk

mendapatkan bentuk dasar kata dengan menghilankan *suffix* (akhiran) (Babu & Kanaga, 2022). Ada beberapa contoh algoritma *stemming* (*stemmer*) yang populer digunakan seperti:

- *Porter stemmer*: pertama kali diusulkan oleh Porter pada tahun 1980 yang diterbitkan dalam penelitiannya. Algoritma ini berbasis atura yang digunakan untuk menghapus akhiran kata.
- *Snowball stemmer*: dikembangkan pada tahun 1996 oleh Porter. Stemmer berbasis aturan yang merupakan pengembangan dari *porter*.
- *Wordnet Lemmatizer*: dikembangkan pertama kali pada tahun 1998, merupakan kombinasi dari beberapa *stemmer* (Fellbaum, 1998). Menggunakan pengetahuan tentang morfologi bahasa untuk menghapus kata akhiran.

2.3.4. Feature Extraction

Feature extraction menjadi aspek signifikan dari teteksi kesalahan informasi karena efektivitas algoritma *machine learning* terutama bergantung pada *feature extraction* (Barve et al., 2022). Menurut Verma et al., (2022) pendekatan *feature extraction* digunakan untuk mengekstraksi fitur berharga dari kumpulan data berdimensi tinggi. Sebab untuk memprediksi atau mengoptimalkan sentimen, diperlukan *window extraction* pada karakter tersembunyi dari kata atau kalimat yang ada. Di sisi lain Ressan & Hassan (2022) mengatakan ini adalah proses kerja yang penting. Proses pengurangan 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 (Rustam et al., 2019). 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* (Rustam et al., 2019). Babu & Kanaga, (2022) memaparkan beberapa teknik *feature extraction* sebagai berikut:

a) *Term Frequency Inverse Document Frequency* (TF-IDF): diusulkan pertama kali oleh Jones pada tahun 1972 dan dikembangkan oleh beberapa peneliti seperti SALTON & BUCKLEY. Menghitung seberapa penting suatu kata bagi suatu dokumen dalam sekelompok dokumen yang dilakukan dengan mengalikan dua metrik. Teknik ini memeriksa penampilan kata-kata dalam dokumen dan frekuensi dokumen dalam kebalikan dari kata di satu set dokumen.
b) *Bag-of-words* (BOW): diusulkan pertama kali pada tahun 1954 oleh Harris, dan dikenalkan dengan istilah BOW pada tahun 1972 oleh Manning et al., 2002. Merupakan representasi teks yang menggambarkan kata-kata dalam dokumen. Melibatkan dua hal yakni kosakata kata yang dikenal dan kata-kata yang dikenal kehadiran totalnya.

c) *Word2Vec*: dikenalkan pada tahun 2013 oleh Mikolov et al., dan Mikolov, Sutskever, et al., model jaringan syaraf (neural *network*) ini akan digunakan untuk mempelajari asosiasi kata dari kospus teks yang besar. Sebuah model dapat mendeteksi sinonim atau menyarankan kata tambahan untuk dijadikan kalimat, setelah dilatih.

d) *Glove*: pertama kali dikenalkan pada tahun 2014 oleh Pennington et al. Termasuk ke dalam algoritma *unsupervised learning* dengan memetakan kata ke dalam ruang bermakna untuk mendapatkan representasi vektor kata di mana kesamaan semantic terkait dengan jarak setiap kata.

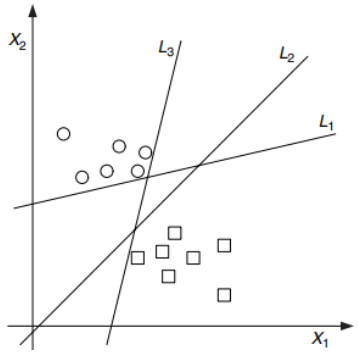
e) N-*Gram*: dikenalkan pertama kali oleh Zipf, 1999 dan dikembangkan oleh beberapa peneliti termasuk Daniel Jurafsky & James H. Martin, 2006. Merupakan salah satu jenis model *probabilistic language* (bahasa probabilitas). Berfungsi untuk memprediksi item berikutnya dalam urutan seperti itu dalam bentuk model *Markov* (*n* 1). Dibangun untuk menghitung seberapa sering urutan kata muncul dalam teks korpus dan kemudia memperkirakan probabilitasnya.

2.3.5. Sumpport Vector Machines (SVMs)

Support vector machines (SVM) adalah milik keluarga model *generalized linear* (linier umum) yang mencapai keputusan klasifikasi atau regresi berdasarkan nilai kombinasi fitur linier. Beberapa juga mengakatan termasuk ke dalam keluarga "metode kernel". SVM juga termasuk kedalam *supervised learning methods* yang menghasilkan fungsi pemetaan input-ouput dari serangkaian data pelatihan berlabel. Fungsi pemetaan dapat berupa fungsi klasifikasi (digunakan untuk

mengkategorikan data input) atau fungsi regresi (digunakan untuk memperkirakan output yang diinginkan) (David L. Olson & Delen, 2008).

Umumnya, banyak pengkasifikasi linier (*hyperplanes*) yang dapat memisahkan data menjadi beberapa kelas. SVM mengklasifikasikan data sebagai bagian dari proses *machine learning*, yang "belajar" dari kasus historikal yang dipresentasikan sebagai poin data. Dimana titik data ini mungkin memiliki lebih dari dua dimensi.



Gambar 2. 1 Contoh Pengklasifikasi Linier

2.3.6. Confusion Matrix

Confusion matrix merupakan sebuah alat untuk mengevaluasi kinerja algoritma *machine learning* yang berisi informasi tentang klasifikasi dan prediksi aktual. Ada empat indikator yang diukur di dalamnya: *accuracy*, *precision*, *recall* dan *F1-Score* (Prastyo et al., 2020). Pada evaluasi klasifikasi ada 4 kemungkinan yang bisa terjadi dari hasil klasifikasi satu data. Bila data positif dan prediksi positif akan dihitung sebagai *true positive* dan jika data positif dan prediksi negatif maka akan dihitung sebagai *false negative*. Pada data negatif jika prediksi negatif makan

akan dihitung sebagai *true negative* sedangkan jika prediksi positif maka akan dihitung sebagai *false positive* (Suryati et al., 2023).

Tabel 2. 1 *Confusion Matrix*

Actual	Prediction	
	Positif	Negatif
Positif	True Positive (TP)	True Negative (TN)
Negatif	False Positive (FP)	False Negative (FN)

a) *Precision*

Precision adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban *precision*. Dihitung dengan rumus seperti pada persamaan (1).

$$Precission = \frac{TP}{(TP+FP)} \text{ ……..……..…….……(1)}$$

b) *Recall*

Recall merupakan perhitungan keakuratan prediksi yang digunakan sebagai ukuran tingkat keberhasilan sistem dalam menemukan sebuah informasi. Dapat dihitung melalui rumus seperti pada persamaan (2).

$$Recall = \frac{TP}{(TP+FN)} \text{ ……..……..……..…….……...(2)}$$

c) *Accuracy*

Accuracy ialah tingkat kedekatan antara nilai prediksi dengan nilai asli atau *actual*. Dihitung dalam rumus seperti pada persamaan (3).

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \text{……..……..….(3)}$$

d) F1-*score*

F1-*score* memperhitungkan kedua *precision* dan *recall*, sehingga lebih sensitive terhadap kesalahan dalam memprediksi kelas minoritas. Dihitung dalam rumus seperti pada persamaan (4).

$$F1 - Measure = 2 \times \frac{precission \times recall}{precission + recall} \quad \text{.....(4)}$$

BAB III

METODE PENELITIAN

3.1. Jenis, Sifat, dan Pendekatan Penelitian

Penelitian ini termasuk kedalam jenis penelitian empiris, mengacu pada penggunaan data yang diperoleh dari penelitian-penelitian sebelumnya dan digunakan untuk menguji hipotesis bahwa *stemming* dan *feature extraction* dapat meningkatkan akurasi algoritma SVM pada analisis sentimen berbasis leksikon.

Sifat dari penelitian ini sendiri adalah ekperimental. Dimana peneliti memanipulasi variabel independent (*stemmer* dan *feature extraction*) untuk mengetahui pengaruh dari keduanya terhadap variabel dependent (kinerja algoritma *SVM*). Di lain sisi, penelitian ini memiliki sifat deskriptif sebab data yang diperoleh dan diolah dibandingkan dengan penelitian sebelumnya.

Pendekatan kuantitatif digunakan pada penelitian ini, sesuai dengan hipotesis akan mengukur kinerja dari algoritma SVM dengan implementasi *stemmer* dan *feature extraction*. Yang dilakukan secara sistematis mengikuti rencana yang terdefinisi dengan baik untuk mengumpulkan dan menganalisis data dengan variabel yang diajukan untuk memastikan validitas dan realibilitas hasil temuan.

3.2. Metode Pengumpulan Data

Sumber data yang dibutuhkan pada penelitian ini memanfaatkan dua metode utama untuk mengumpulkan data, yakni:

a) *Literature review*: penelitian ini melibatkan pengumpulan data dengan analisis *research paper* sebelumnya yang memiliki keterkaitan topik yang diangkat khususnya analisis sentimen. Data yang didapatkan dapat berupa data publik yang tersedia pada media publikasi, *repository* publik dan lain sebagainya.

b) *Secondary data analysis*: ini melibatkan penggunaan data yang telah dikumpulkan oleh peneliti-peneliti sebelumnya. Data yang didapatkan dapat berupa arsip yang tersedia secara publik dan diperbolehkan untuk digunakan, biasanya tedapat pada *repository GITHUB*, *Driver* dan lain sebagainya.

3.3. Metode Analisis Data

Untuk mencapai tujuan dan membuktika hipotesa dari penelitian ini, maka beberapa tahapan metode analisis data dilakukan. Dengan catatan untuk mengetahui dampak *stemmer* dan *feature extraction* yang digunakan pada kinerja algoritma SVM dalam analisis sentimen berbasis leksikon. Berikut adalah tahapan-tahapan yang akan dilakukan.

Langkah pertama dalam proses analisis data yakni melakukan *preprocessing* data. *Preprocessing* data melibatkan, *case folding, cleaning, tokenizing, stopwords removal* dan kemudia *stemming*. Pada proses *stemming* beberapa algoritma digunakan yakni *Porter Stemmer*, *Snowball Stemmer*, dan *WordNet Lemmatizer*.

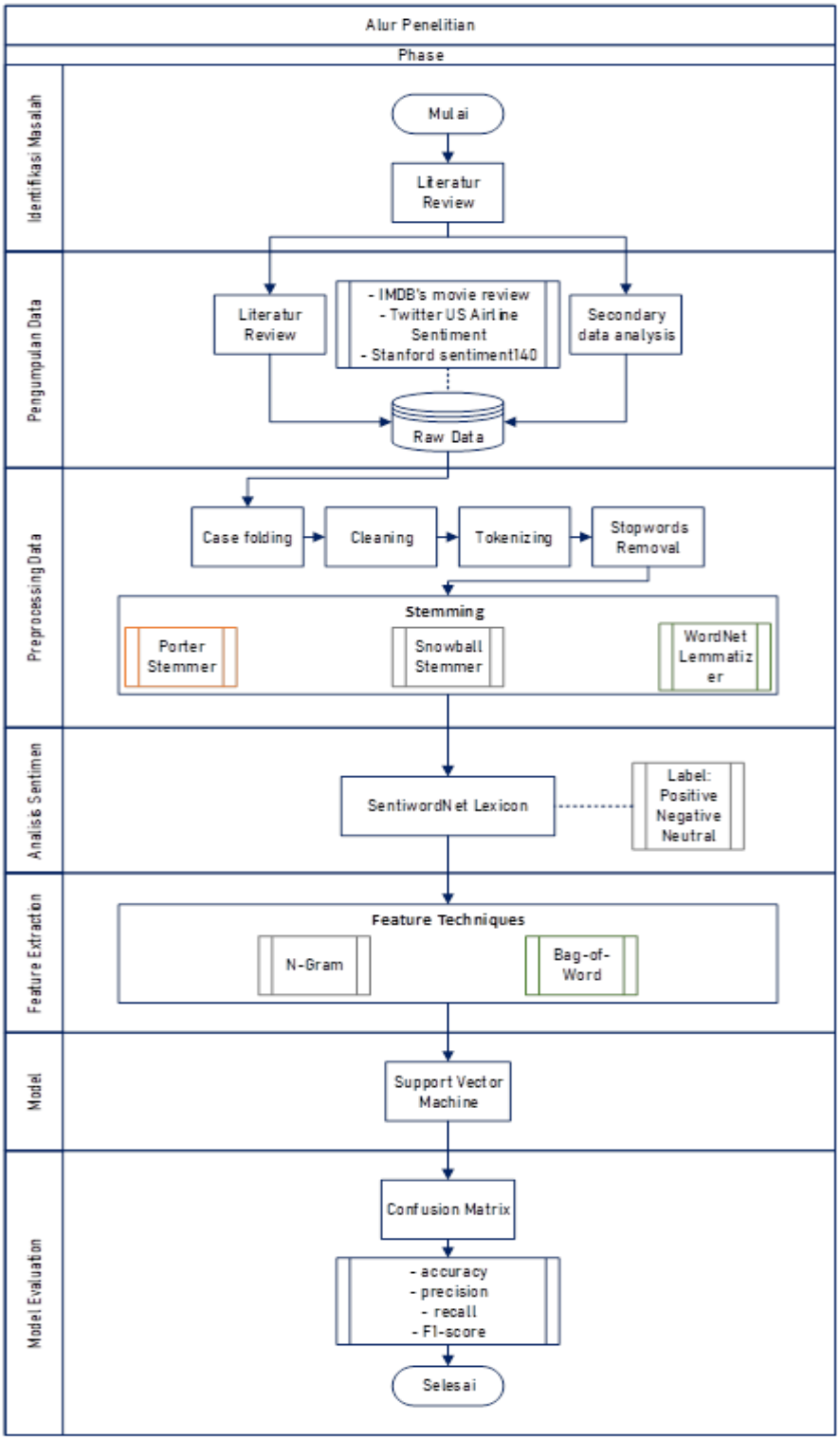
Berikutnya adalah prose klasifikasi data menggunakan metode leksikon. Ini bertujuan untuk mengukur polaritas dari data yang kemudian akan dijadikan label kelas berupa positif, negatif maupun netral. Dan pada algoritma SVM akan digunakan sebagai nilai Y.

Feature extraction data merupakan langkah ketiga dari analisis data. Proses ini dilakukan untuk kemudian digunakan sebagai variable independent pada algoritma SVM. Feature extraction yang diimplementasi pada penelitian ini adalah *N-grams,* dan *Bag-of-Words* (BoW)

Langkah terakhir adalah melatih dan mengevaluasi algortima SVM. Algoritma SVM dilatih pada beberapa data. Yang kemudia kinerja algortima SVM dievaluasi pada serangkaian data uji. Hasil yang akan dihitung yakni *accuracy, precision, recall* dan *F1-Score*. Algoritma SVM akan diimplementasi pada beberapa gabungan antara *stemmer* dan *feature extraction* untuk mengetahui bagunan terbaik dari keduanya

3.4. Alur Penelitian

Untuk menyelesaikan penelitian ini dibutuhkan langkah-langkah yang harus diselesaikan. Adapun langkah atau tahap yang dilakukan pada penelitian ini dapat dilihat pada Gambar 3.1 yang dipaparkan sebagai berikut:



Gambar 3. 1 Alur Penelitian

3.4.1. *Literature Review*

Literature review atau tinjauan literatur dilakukan untuk mendidentifikasi penelitian yang ada dengan topik serupa dalam kasus ini penggunaan SVM pada analisis sentimen berbasis leksikon. Dengan ini dapat membantu peneliti untuk mengidentifikasi kesenjangan penelitian dan merumuskan pertanyaan tentang penelitian yang akan dilakukan.

3.4.2. Pengumpulan Data

Seperti yang telah dipaparkan pada bagian "Metode Pengumpulan Data", penelitian ini menggunakan data public yang sudah digunakan pada penelitian-penelitian sebelumnya. Ini dilakukan untuk menghemat waktu penelitian karena tidak perlu melakukan pelabelan secara manual dan melibatkan pakar bahasa. Disisi lain untuk memperkuat validitas dan kredibilitas dari temuan penelitian, penelitian ini menggunakan beberapa dataset. Dengan membandingkan hasil dari beberapa sumber data, akan dapat mengidentifikasi tren, pola dan anomali yang kemungkinan tidak akan terlihat jika hanya menggunakan satu dataset.

Penggunaan beberapa dataset juga didasarkan pada pengurangan bias. Karena setiap dataset memiliki bias tersendiri. Hal ini akan membantu memastikan bahwa pemilihan model akan lebih objektif dan dapat diterapkan lebih luas.Seperti yang diperlihatkan pada Tabel 3. 1, dipilihnya 3 dataset tersebut adalah sebagai bencmarking. Dimana 3 dataset yang dipilih telah digunakan secara luas sebagai bencmark untuk penelitian analisis sentimen seperti oleh Tan et al., (2022), Tabinda Kokab et al., (2022), Saad, (2020) dan Harjule et al., (2020). Dengan

membandingkan hasil dari penelitian ini dengan penelitian-penelitian sebelumnya peneliti dapat menilai kinerja dari model yang digunakan. Tabel 3.1 menampilkan statistik dari dataset yang digunakan pada penelitian ini.

Tabel 3. 1 Statistik Dataset

Dataset	Source	Clasess
IMDB's movie review [1]	(Shaddeli et al., 2022)	Positive, Negative
Twitter US Airline Sentiment [2]	(AlBadani et al., 2022)	Positive, Negative, Neutral
Stanford sentiment140 [3]	(Ahmed et al., 2022)	Positive, Negative

3.4.3. *Preprocessing* Data

Tahap ini dilakukan untuk memproses data (teks) mentah (tidak terstruktur) menjadi teks yang terstruktur, dengan tujuan memaksimalkan hasil dari analisis sentimen. Telah dijelaskan pada bagian "Metode Analisis Data" proses ini melibatkan *case folding, cleaning, tokenizing, stopwords removal* dan kemudia *stemming*. Berikut penjelasan dari langkah-langkah pada *preprocessing*:

a) *Case Folding*

Langkah ini berfungsi untuk mengubah huruf besar atau kapital menjadi huruf kecil atau *lowercase*.

b) *Cleaning*

Cleaning bertujuan untuk membersihkan hal tidak penting pada analisis sentimen seperti karakter, nomor, tanda baca, w*hite space* dan *single char*.

[1] Available online: https://ai.stanford.edu/$\sim$7amaas/data/sen-timent/
[2] Available online: https://www.kaggle.com/crowdflower/twitter-airline-sentiment
[3] Available online: https://help.senti-ment140.com/site-functionality

c) *Tokenizing*

Tokenisasi atau *tokenizing* mengacu kepada pemisahan document teks (kalimat) menjadi unit-unit kecil. Dimana unit (kata) disebut dengan token (Shamrat et al., 2021). Secara umum tokenisasi merupakan proses pemecahan kalimat menjadi kata yang dalam kasus ini setiap kata pada kalimat ditampung kedalam bentuk *array*.

d) *Stopword Removal*

Fungsi dari proses ini ialah membuang kata dasar atau kata yang ada pada *list stop* yang memiliki frekuensi kemunculan tinggi, misalnya kata penghubung seperti “will”, “but”, “or”, “and” dan “for” (Nurcahyawati & Mustaffa, 2023).

e) *Stemming*

Stemming merupakan teknik untuk mengubah kata menjadi kata dasar dengan menghilangkan imbuhan baik akhiran atau awalan yang melekat pada kata (Nurcahyawati & Mustaffa, 2023). Seperti pada kata “helpful” menjadi “help”.

Pada proses *stemming* beberapa algoritma digunakan yakni *Porter Stemmer*, *Snowball Stemmer*, dan *WordNet Lemmatizer*. Beberapa *stemming* ini digunakan untuk nantinya dibandingkan satu sama lain dengan gabungan *feature extraction pada* proses evaluasi menggunakan algoritma SVM. Pemilihan ketiga *stemmer* ini didasarkan pada alasan-alasan berikut:

- Efektivitas dan kesederhanaan: *Porter Stemmer*, *Snowball Stemmer* dikenal dengan kesederhanaan dan kemudahan dalam implementasinya. Algoritma keduanya mudah dipahami dan diimplementasikan, walau dengan sumber daya yang terbatas. Di sisi lain, *WordNet Lemmatizer*

menawarkan pendekatan *lemmatization* lebih canggih dengan memanfaatkan informasi leksikal dan morfologi dari wordnet.

- Kesesuaian dengan dataset: Berdasarkan dataset yang akan digunakan dimana mengandung teks informal dari komentar ataupun *tweet* dengan banyak singkatan dan kata tidak beraturan. Ketiga metode dapat diandalkan untuk mengurangi *noise* dan meningkatkan konsentrasi representasi kata, yang dapat menguntungkan tugas leksikon sebagai analisis sentimen.

3.4.4. Analisis sentimen (*lexicon-based*)

Tahap ini bertujuan menentukan apakah kalimat atau dokumen berlabel positif, negatif atau netral. Sebelum memberi label pada kalimat dihitung nilai polaritasnya terlebih dahulu. Penelitian ini menggunakan leksikon untuk analisis sentimen dengan *dictionary-based approach*. Dengan menggunakan *Dictionary-Based Approach* atau pendekatan berbasis kamus peneliti diberikan kemudahan karena tidak perlu membuat korpus sendiri. Kamus yang dapat digunakan seperti *WordNet*, *SentiWordNet*, *Affin* dan lain sebagainya, penelitian ini menggunakan *SentiWordNet*. Untuk pemberian label sendiri digunakan perhitungan seperti dibawah:

Positif: nilai polaritas > 0

Negatif: nilai polaritas < 0

Netral: nilai polaritas = 0

3.4.5. *Feature Extraction*

Feature extraction digunakan untuk mengubah data menjadi format yang kemudian dapat digunakan oleh algoritma *machine learning* dalam penelitian ini adalah SVM. Proses ini akan merubah teks disiapkan untuk data tabular. *Feature extraction* (*N-grams*, dan *Bag-of-Words* (BoW)) diimplemtasi untuk nantinya dibandingkan satu sama lain dengan gabungan *stemmer* pada proses evaluasi menggunakan algoritma SVM.

Pemilihan kedua metode ini didasarkan pada relevansi dengan penelitian penelitian terbaru tentang analisis sentimen atau text mining seperti penelitian oleh Semary et al., (2024), Qi & Shabrina, (2023) dan Rustam et al., (2021) yang masih menggunakan keduanya. Meskipun sudah ada banyak *feature extraction* terbaru seperti *doc2vec*, *word2vec* dan yang lain. Selebihnya *N-grams*, dan *Bag-of-Words* (BoW) memiliki keunggulan sebagai berikut:

- Kemampuan Interpretasi: N-grams dan BoW menawarkan interpretasi yang lebih mudah dibandingkan metode feature extraction yang menggunakan embedding seperti word2vec maupun doc2vec. Kemampuan interpretasi ini sangatlah penting, terutama dalam penelitian ilmiah di mana transparansi dan pemahaman mendalam tentang hasil sangatlah penting.
- Efisiensi komputasi: N-grams dan BoW secara umum lebih hemat dalam komputasi dibandingkan dengan metode embedding kata, terutama dengan dataset besar. Ini menjadi hal yang lebih penting karena pada penelitian ini menggunakan metode hybrid dimana ada beberapa metode dan parameter

yang digunakan. Di mana runtime yang lebih cepat memungkinkan iterasi yang lebih banyak dan optimasi pada model lebih efektif.

- Kinerja yang kompetitif: Di sisi lain keduanya menunjukkan dengan teknik augmentasi data dan pemilihan fitur yang tepat dapat mencapai hasil yang sebanding, bahkan lebih baik dibandingkan dengan metode embedding kata, seperti pada penelitian oleh Shuai et al., (2022), Kalangi et al., (2021) dan Rajesh & Suseendran, (2020). Ini menunjukkan keduanya masih dapat menjadi pilihan yang efektif.
- Kemudahan implementasi dan kesederhanaan: Keduanya juga merupakan metode yang relatif sederhana dan mudah diimplementasi dibandingkan yang lain karena lebih kompleks. Keuntungan ini akan menghemat waktu dalam penelitian karena menggunakan beberapa skenario.

3.4.6. *Model* (*Support Vector Machine*)

SVM dipilih sebagai model klasifikasi pada penelitian berdasarkan hasil literatur *review* pada beberapa penelitian Obiedat et al., (2022), Barve et al., (2022), Kumar & Subba, (2020), Fikri & Sarno, (2019) dan Rahat et al., (2019), dan pertimbangan cermat terhadap beberapa faktor, yakni:

- SVM dikenal dengan kemampuan generalisasi tinggi, yang memiliki arti model dapat belajar dari data latih dan memprediksi secara akurat pada data test.
- Analisis sentimen seringkali melibatkan dataset yang tidak imbang, di mana jumlah contoh positif dan negatif atau netral tidak sama. SVM secara

alami mampu menangani ketidakseimbangan data ini, menghasilkan klasifikasi yang akurat bahkan ketika salah satu kelas lebih dominan.

- SVM mampu menangani data berdimensi tinggi dengan efektif. Ini sangat dibutuhkan pada analisis sentimen yang seringkali melibatkan data berdimensi tinggi, dimana terdapat banyak fitu yang diekstrak dari teks.
- Demi mencegah penurunan performa pada dataset yang tidak ada selama penelitian. SVM secara inheren memiliki kemampuan untuk meminimalisir overfitting, yaitu kondisi di mana model terlalu terlatih pada data pelatihan dan tidak dapat bekerja dengan baik pada data baru.

Setelah melakukan *preprocessing*, analisis sentimen dengan *lexicon-based*, dan *feature extraction*, kemudian *model training* SVM dilakukan. Langkah pertama pada proses ini adalah menentukan vairbel X dan Y dimana X adalah hasil stemming dan Y adalah label dari polaritas leksikon *SentiWordNet*. Selanjutnya data dibagi menjadi data *test* dan data *train* dengan perbandingan 80 / 20. Kemudian *feature extraction* dilatih pada algoritma SVM. Proses ini akan dilakukan secara berulang dengan penggabungan antara jenis *stemmer* dan *feature extraction* yang kemudia performa dari model akan dievaluasi menggunakan *Confusion Matrix* dan dibandingkan. Dan dilakukan berulang pada data yang berbeda.

3.4.7. *Model Evaluation* (*Confusion Matrix*)

Model divaluasi menggunakan *confusion matrix*. *Confusion matrix* merupakan sebuah alat untuk mengevaluasi kinerja algoritma *machine learning* yang berisi informasi tentang klasifikasi dan prediksi aktual. Ada empat indikator

yang diukur di dalamnya: *accuracy*, *precision*, *recall* dan *F1-Score*. Gambar 3.1 dibawah menunjukkan 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 3. 2 *Confusion matrix* (Prastyo et al., 2020)

Adapun perhitungan dari 4 indikator adalah sebagai berikut:

$$Precission = \frac{TP}{(TP+FP)} \quad \text{.....(1)}$$

$$Recall = \frac{TP}{(TP+FN)} \quad \text{.....(2)}$$

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \quad \text{.....(3)}$$

$$F1 - Measure = 2 \times \frac{precission \times recall}{precission + recall} \quad \text{.....(4)}$$

BAB IV

HASIL DAN PEMBAHASAN

4.1. Pengumpulan Data

Penelitian ini mendapatkan data berdasarkan dua metode pengumpulan data yang digunakan yakni *literature review* dan *secondary data analysis*. Dari kedua metode tersebut penelitian kali ini menggunakan 3 dataset yang berbeda, dengan tujuan untuk memperkuat hasil eksperimen. Pada Tabel 4.1 ditunjukkan satastik jumlah dari masing-masing dataset yang digunakan.

Tabel 4. 1 Statistik Jumlah Dataset

Dataset	Sumber	Classes	Jumlah
IMDB's movie review (Dataset 1)	(Shaddeli et al., 2022)	Positive, Negative	50000
Twitter US Airline Sentiment (Dataset 2)	(AlBadani et al., 2022)	Positive, Negative, Neutral	14640
Stanford sentiment140 (Dataset 3)	(Ahmed et al., 2022)	Positive, Negative	1600000

Dataset yang digunakan pada penelitian ini sudah dilengkapi oleh variabel sentimen dimana pada dataset 1 dan 3 terdapat 1 class dan pada dataset 2 berisi 3 class. Setiap class pada masing-masing dataset memiliki jumlah yang berbeda. Statistik perbedaan jumlah class dari ketiga dataset ditunjukkan pada Tabel 4.2.

Tabel 4. 2 Statistik Jumlah Class Sentimen Dataset

Dataset / Classes	Positive	Negative	Neutral
IMDB's movie review (Dataset 1)	25000	25000	-
Twitter US Airline Sentiment (Dataset 2)	2363	9178	3099
Stanford sentiment140 (Dataset 3)	800000	800000	-

4.1.1. Memilih Variabel

Dari ketiga dataset yang digunakan, masing-masing memiliki variabel yang berbeda. Maka peneliti hanya memilih variabel penting yang dibutuhkan pada penelitian. Pada Tabel 4.3 ditunjukkan variabel awal dan variabel akhir pada setiap dataset.

Tabel 4. 3 Variabel Dataset

Dataset	Variabel Awal	Variabel Akhir
IMDB's movie review (Dataset 1)	review, sentiment	review, sentiment
Twitter US Airline Sentiment (Dataset 2)	tweet_id, airline_sentiment, airline_sentiment_confidence, negativereason, negativereason_confidence, airline, airline_sentiment_gold, name, negativereason_gold, retweet_count, text, tweet_coord, tweet_created, tweet_location, user_timezone	airline_sentiment, text
Stanford sentiment140 (Dataset 3)	sentiment, ids, date, flag, user, text	sentiment, text

Variabel akhir yang digunakan merupakan variable inti yang dibutuhkan pada penelitian. Variabel review pada dataset 1 dan variabel text pada dataset 2 dan 3 merupakan variabel yang berisi teks akan digunakan bertahap dalam penelitian. Dan variabel airline_sentiment pada dataset 2 dan sentiment pada dataset 1 dan 3 merupakan variabel yang berisi *class* (asli) sentiment pada setiap teks. Penelitian hanya menyisakan dua variabel dari setiap dataset dikarenakan yang akan digunakan pada pengujian hanya variabel yang berisi data teks dan *class*.

Setelah data terkumpul dan dipilah variabel yang akan digunakan, tahap selanjutnya adalah *preprocesing*. Tahap ini berisi beberapa langkah, dipaparkan pada bagian 4.2.

4.2. *Preprocessing* Data

Tahapa ini memiliki 5 langkah yakni *case folding*, *cleaning*, *tokenizing*, *stopword removal* dan *stemming*. Pada *stemming* akan dilakukan dengan 3 jenis *stemmer* yakni *porter stemmer*, *snowball stemmer* dan *wordnet lemmatizer*. Preprocessing dilakukan untuk mempersiapkan data teks, karena data bersih dan terstruktur akan membantu meningkatkan akurasi pada model yang digunakan. Tahap ini dilakukan pada semua dataset yang digunakan pada penelitian. Berikut adalah hasil dari setiap langkah pada tahapan *preprocessing*.

4.2.1. *Case Folding*

Langkah ini dilakukan untuk mengubah setiap huruf yang terdapat pada teks dari huruf kapital menjadi huruf kecil (*lowercase*). Langkah ini disertakan pada *preprocessing* karena data yang ada tidak selalu terstruktur dan konsisten, serta untuk mengurangi variasi dan memastikan bahwa kata-kata dengan huruf besar atau kecil yang sama dianggap identik. Pada Tabel 4.4, 4.5 dan 4.6 merupakan hasil *case folding* pada tiga dataset.

Tabel 4. 4 Hasil *Case Folding* Dataset 1

index	review	text_cf
49995	I thought this movie did a down right good job...	i thought this movie did a down right good job...
49996	Bad plot, bad dialogue, bad acting, idiotic di...	bad plot, bad dialogue, bad acting, idiotic di…
49997	I am a Catholic taught in parochial elementary...	i am a catholic taught in parochial elementary...
49998	I'm going to have to disagree with the previou...	i'm going to have to disagree with the previou…
49999	No one expects the Star Trek movies to be high...	no one expects the star trek movies to be high...

Tabel 4. 5 Hasil *Case Folding* Dataset 2

index	text	text_cf
14635	@AmericanAir thank you we got on a different f...	@americanair thank you we got on a different f...
14636	@AmericanAir leaving over 20 minutes Late Flig...	@americanair leaving over 20 minutes late flig...
14637	@AmericanAir Please bring American Airlines to...	@americanair please bring american airlines to...
14638	@AmericanAir you have my money, you change my ...	@americanair you have my money, you change my ...
14639	@AmericanAir we have 8 ppl so we need 2 know h...	@americanair we have 8 ppl so we need 2 know h...

Tabel 4. 6 Hasil *Case Folding* Dataset 3

index	text	text_cf
1599995	Just woke up. Having no school is the best fee...	just woke up. having no school is the best fee...
1599996	TheWDB.com - Very cool to hear old Walt interv...	thewdb.com - very cool to hear old walt interv...
1599997	Are you ready for your MoJo Makeover? Ask me f...	are you ready for your mojo makeover? ask me f...
1599998	Happy 38th Birthday to my boo of alll time!!! ...	happy 38th birthday to my boo of alll time!!! ...
1599999	happy #charitytuesday @theNSPCC @SparksCharity...	happy #charitytuesday @thenspcc @sparkscharity...

4.2.2. *Cleaning*

Selanjutnya pada *preprocessing* dilakukan *cleaning* (pembersihan). Tujuan utama dari *cleaning* ialah untuk mempermudah proses analisis dengan meningkatkan kualitas data. Dimana fungsi utama dari *cleaning* adalah membersihkan *noise*. *Noise* merupakan data yang tidak relevan atau dianggap mengganggu untuk analisis, seperti karakter, nomor, tanda baca, *white space* dan

single char. *Cleaning* teks dapat mengurangi miss-klasifikasi dan meningkatkan akurasi analisis. Berikut adalah hasil cleaning pada setiap dataset.

Tabel 4. 7 Hasil *Cleaning* Dataset 1

index	text_cf	text_clean
49995	i thought this movie did a down right good job...	thought this movie did down right good job i...
49996	bad plot, bad dialogue, bad acting, idiotic di…	bad plot bad dialogue bad acting idiotic direc...
49997	i am a catholic taught in parochial elementary...	am catholic taught in parochial elementary s...
49998	i'm going to have to disagree with the previou…	im going to have to disagree with the previous...
49999	no one expects the star trek movies to be high...	no one expects the star trek movies to be high...

Tabel 4. 8 Hasil *Cleaning* Dataset 2

index	text_cf	text_clean
14635	@americanair thank you we got on a different f...	thank you we got on different flight to chicago
14636	@americanair leaving over 20 minutes late flig...	leaving over minutes late flight no warnings o...
14637	@americanair please bring american airlines to...	please bring american airlines to
14638	@americanair you have my money, you change my ...	you have my money you change my flight and don...
14639	@americanair we have 8 ppl so we need 2 know h...	we have ppl so we need know how many seats are...

Tabel 4. 9 Hasil *Cleaning* Dataset 3

index	text_cf	text_clean
1599995	just woke up. having no school is the best fee...	just woke up having no school is the best feel...
1599996	thewdb.com - very cool to hear old walt interv...	thewdbcom very cool to hear old walt interviews
1599997	are you ready for your mojo makeover? ask me f...	are you ready for your mojo makeover ask me fo...

Tabel 4.9 (Lanjutan)

index	text_cf	text_clean
1599998	happy 38th birthday to my boo of alll time!!! ...	happy th birthday to my boo of alll time tupac...
1599999	happy #charitytuesday @thenspcc @sparkscharity...	happy

4.2.3. *Tokenizing*

Tokenizing atau tokenisasi merupakan proses pemecahan kalimat menjadi kata yang dalam kasus ini setiap kata atau unit yang disebut token pada kalimat ditampung kedalam bentuk *array*. Tokenizing dapat membantu dalam langkah-langkah selanjutnya seperti *stopword removal* dan *stemming*. Begitu juga pada proses analisis sentiment. Karena dengan dengan mengubah kalimat menjadi unit atau token akan mempermudah pencarian kata. Pada Tabel 4.10 sampai Tabel 4.12 merupakan hasil *tokenizing* pada setiap dataset.

Tabel 4. 10 Hasil *Tokenizing* Dataset 1

index	text_clean	text_token
49995	thought this movie did down right good job i...	[thought, this, movie, did, down, right, good,...
49996	bad plot bad dialogue bad acting idiotic direc...	[bad, plot, bad, dialogue, bad, acting, idioti...
49997	am catholic taught in parochial elementary s...	[am, catholic, taught, in, parochial, elementa...
49998	im going to have to disagree with the previous...	[im, going, to, have, to, disagree, with, the,...
49999	no one expects the star trek movies to be high...	[no, one, expects, the, star, trek, movies, to...

Tabel 4. 11 Hasil *Tokenizing* Dataset 2

index	text_clean	text_token
14635	thank you we got on different flight to chicago	[thank, you, we, got, on, different, flight, t...
14636	leaving over minutes late flight no warnings o...	[leaving, over, minutes, late, flight, no, war...
14637	please bring american airlines to	[please, bring, american, airlines, to]

Tabel 4.11 (Lanjutan)

index	text_clean	text_token
14638	you have my money you change my flight and don...	[you, have, my, money, you, change, my, flight...
14639	we have ppl so we need know how many seats are...	[we, have, ppl, so, we, need, know, how, many,...

Tabel 4. 12 Hasil *Tokenizing* Dataset 3

index	text_clean	text_token
1599995	just woke up having no school is the best feel...	[just, woke, up, having, no, school, is, the, ...
1599996	thewdbcom very cool to hear old walt interviews	[thewdbcom, very, cool, to, hear, old, walt, i...
1599997	are you ready for your mojo makeover ask me fo...	[are, you, ready, for, your, mojo, makeover, a...
1599998	happy th birthday to my boo of alll time tupac...	[happy, th, birthday, to, my, boo, of, alll, t...
1599999	happy	[happy]

4.2.4. *Stopword Removal*

Langkah ini ditujukan untuk membuang kata dasar atau kata yang ada pada *list stop* yang memiliki frekuensi kemunculan tinggi, misalnya kata penghubung seperti “will”, “but”, “or”, “and” dan “for”. *Stopword removal* dapat mengurangi demensi data teks, dimana hal ini dapat meningkatkan kecepatan dan efesiensi analisis nantinya. Berikut pada Tabel 4.13 sampai Tabel 4.15 adalah hasil *stopword removal*.

Tabel 4. 13 Hasil *Stopword Removal* Dataset 1

index	text_token	text_swr
49995	[thought, this, movie, did, down, right, good,...	[thought, movie, right, good, job, wasnt, crea...
49996	[bad, plot, bad, dialogue, bad, acting, idioti...	[bad, plot, bad, dialogue, bad, acting, idioti...
49997	[am, catholic, taught, in, parochial, elementa...	[catholic, taught, parochial, elementary, scho...
49998	[im, going, to, have, to, disagree, with, the,...	[im, going, disagree, previous, comment, side,...
49999	[no, one, expects, the, star, trek, movies, to...	[one, expects, star, trek, movies, high, art, ...

Tabel 4. 14 Hasil *Stopword Removal* Dataset 2

index	text_token	text_swr
14635	[thank, you, we, got, on, different, flight, t...	[thank, got, different, flight, chicago]
14636	[leaving, over, minutes, late, flight, no, war...	[leaving, minutes, late, flight, warnings, com...
14637	[please, bring, american, airlines, to]	[please, bring, american, airlines]
14638	[you, have, my, money, you, change, my, flight...	[money, change, flight, dont, answer, phones, ...
14639	[we, have, ppl, so, we, need, know, how, many,...	[ppl, need, know, many, seats, next, flight, p...

Tabel 4. 15 Hasil *Stopword Removal* Dataset 3

index	text_token	text_swr
1599995	[just, woke, up, having, no, school, is, the, ...	[woke, school, best, feeling, ever]
1599996	[thewdbcom, very, cool, to, hear, old, walt, i...	[thewdbcom, cool, hear, old, walt, interviews]
1599997	[are, you, ready, for, your, mojo, makeover, a...	[ready, mojo, makeover, ask, details]
1599998	[happy, th, birthday, to, my, boo, of, alll, t...	[happy, th, birthday, boo, alll, time, tupac, ...
1599999	[happy]	[happy]

4.2.5. *Stemming*

Stemming merupakan langkah terakhir pada tahapan *preprocessing* data. Dimana langkah ini bertujuan untuk mengubah kata yang ada menjadi kata dasar dengan menghilangkan imbuhan baik akhiran atau awalan yang melekat pada kata. Seperti yang telah dipaparkan pada bagian alur penelitian, algoritma atau jenis *stemming* yang digunakan pada penelitian ini adalah *Porter Stemmer*, *Snowball Stemmer*, dan *WordNet Lemmatizer*. Pada Tabel 4.16 sampai Tabel 4.18 merupakan hasil *stemming* menggunakan *porter stemmer*, Tabel 4.19 sampai Tabel 4.21 merupakan hasil *stemming* menggunakan *snowball stemmer* dan pada Tabel 4.22 sampai Tabel 4.24 merupakan hasil *stemming* menggunakan *wordnet lemmatizer*.

- *Porter Stemmer*

Tabel 4. 16 Hasil *Porter Stemmer* Dataset 1

index	text_swr	text_porter
49995	[thought, movie, right, good, job, wasnt, crea...	thought movi right good job wasnt creativ orig...
49996	[bad, plot, bad, dialogue, bad, acting, idioti...	bad plot bad dialogu bad act idiot direct anno...
49997	[catholic, taught, parochial, elementary, scho...	cathol taught parochi elementari school nun ta...
49998	[im, going, disagree, previous, comment, side,...	im go disagre previou comment side maltin one ...
49999	[one, expects, star, trek, movies, high, art, ...	one expect star trek movi high art fan expect ...

Tabel 4. 17 Hasil *Porter Stemmer* Dataset 2

index	text_swr	text_porter
14635	[thank, got, different, flight, chicago]	thank got differ flight chicago
14636	[leaving, minutes, late, flight, warnings, com...	leav minut late flight warn commun minut late ...
14637	[please, bring, american, airlines]	pleas bring american airlin
14638	[money, change, flight, dont, answer, phones, ...	money chang flight dont answer phone suggest m...
14639	[ppl, need, know, many, seats, next, flight, p...	ppl need know mani seat next flight plz put us…

Tabel 4. 18 Hasil *Porter Stemmer* Dataset 3

index	text_swr	text_porter
1599995	[woke, school, best, feeling, ever]	woke school best feel ever
1599996	[thewdbcom, cool, hear, old, walt, interviews]	thewdbcom cool hear old walt interview
1599997	[ready, mojo, makeover, ask, details]	readi mojo makeov ask detail
1599998	[happy, th, birthday, boo, alll, time, tupac, ...	happi th birthday boo alll time tupac amaru sh...
1599999	[happy]	happi

- *Snowball Stemmer*

Tabel 4. 19 Hasil *Snowball Stemmer* Dataset 1

index	text_swr	text_snowball
49995	[thought, movie, right, good, job, wasnt, crea...	thought movi right good job wasnt creativ orig...
49996	[bad, plot, bad, dialogue, bad, acting, idioti...	bad plot bad dialogu bad act idiot direct anno...

Tabel 4.19 (Lanjutan)

index	text_swr	text_snowball
49997	[catholic, taught, parochial, elementary, scho...	cathol taught parochi elementari school nun ta...
49998	[im, going, disagree, previous, comment, side,...	im go disagre previous comment side maltin one...
49999	[one, expects, star, trek, movies, high, art, ...	one expect star trek movi high art fan expect ...

Tabel 4. 20 Hasil *Snowball Stemmer* Dataset 2

index	text_swr	text_snowball
14635	[thank, got, different, flight, chicago]	thank got differ flight chicago
14636	[leaving, minutes, late, flight, warnings, com...	leav minut late flight warn communic minut lat...
14637	[please, bring, american, airlines]	pleas bring american airlin
14638	[money, change, flight, dont, answer, phones, ...	money chang flight dont answer phone suggest m...
14639	[ppl, need, know, many, seats, next, flight, p...	ppl need know mani seat next flight plz put us...

Tabel 4. 21 Hasil *Snowball Stemmer* Dataset 3

index	text_swr	text_snowball
1599995	[woke, school, best, feeling, ever]	woke school best feel ever
1599996	[thewdbcom, cool, hear, old, walt, interviews]	thewdbcom cool hear old walt interview
1599997	[ready, mojo, makeover, ask, details]	readi mojo makeov ask detail
1599998	[happy, th, birthday, boo, alll, time, tupac, ...	happi th birthday boo alll time tupac amaru sh...
1599999	[happy]	happi

- Wordnet Lemmatizer

Tabel 4. 22 Hasil *Wordnet Lemmatizer* Dataset 1

index	text_swr	text_lemmatize
49995	[thought, movie, right, good, job, wasnt, crea...	thought movie right good job wasnt creative or...
49996	[bad, plot, bad, dialogue, bad, acting, idioti...	bad plot bad dialogue bad acting idiotic direc..
49997	[catholic, taught, parochial, elementary, scho...	catholic taught parochial elementary school nu...
49998	[im, going, disagree, previous, comment, side,...	im going disagree previous comment side maltin...
49999	[one, expects, star, trek, movies, high, art, ...	one expects star trek movie high art fan expec...

Tabel 4. 23 Hasil *Wordnet Lemmatizer* Dataset 2

index	text_swr	text_lemmatize
14635	[thank, got, different, flight, chicago]	thank got different flight chicago
14636	[leaving, minutes, late, flight, warnings, com...	leaving minute late flight warning communicati...
14637	[please, bring, american, airlines]	please bring american airline
14638	[money, change, flight, dont, answer, phones, ...	money change flight dont answer phone suggesti...
14639	[ppl, need, know, many, seats, next, flight, p...	ppl need know many seat next flight plz put u ...

Tabel 4. 24 Hasil *Wordnet Lemmatizer* Dataset 3

index	text_swr	text_lemmatize
1599995	[woke, school, best, feeling, ever]	woke school best feeling ever
1599996	[thewdbcom, cool, hear, old, walt, interviews]	thewdbcom cool hear old walt interview
1599997	[ready, mojo, makeover, ask, details]	ready mojo makeover ask detail
1599998	[happy, th, birthday, boo, alll, time, tupac, ...	happy th birthday boo alll time tupac amaru sh...
1599999	[happy]	happy

Hasil *stemming* yang ditampilkan menunjukkan beberapa perbedaan pada setiap kata yang dibuah. Baik antara *Porter stemmer* dan *Snowball* atau atara keduanya dengan *Wordnet Lemmatizer*. Ini dikarenakan ketiganya memiliki cara kerja yang berbeda. *Porter Stemmer* menghapus awalan dan akhiran dari kata untuk mendapatkan akar kata. Seperti halnya dengan *Porter stemmer*, *Snowball stemmer* melakukan hal yang sama namun dengan aturan yang sedikit lebih baik dari pada *Porter*. Dan untuk *Wordnet Lemmatizer* sendiri melakukan perubahan kata pada kata dasar berdasarkan konteks dan *part-of-speech*. Dari hasil ketiganya akan menghasilakan analisis sentimen yang berbeda pula, seperti yang dipaparkan pada bagian 4.3.

4.3. Analisis Sentimen (*Lexicon-based*)

Setelah tahap *preprocessing* dilakukan dengan proses *stemming* di akhir yang menghasilkan data tiga hasil *stemming* yang berbeda, proses selanjutkan adalah analisis sentimen berbasis leksikon (*Lexicon-based*). Tahapan ini bertujuan untuk menentukan apakah dokumen atau kalimat yang ada berlabel positif, negatif, ataupun netral. Untuk mendapatkan label tersebut kalimat akan dihitung nilai polaritasnya. Sebagaimana yang telah dipaparkan pada bagian 3.4.4, penelitian ini menggunakan *SentiWordNet* (SWN) *lexical resource* sebagai sumber kamus leksikal. Dan untuk pelabelan pada teks menggunakan perhitungan seperti berikut:

Positif: nilai polaritas > 0

Negatif: nilai polaritas < 0

Netral: nilai polaritas = 0

Berikut hasil dari analisis sentiment menggunakan *SWN lexical resource*, Pada Tabel 4.25 - Tabel 4.27 merupakan hasil leksikon pada *porter stemmer*, Tabel 4.28 - Tabel 4.30 merupakan hasil leksikon pada *snowball stemmer* dan pada Tabel 4.31 - Tabel 4.33 merupakan hasil leksikon pada *wordnet lemmatizer*.

- *Porter Stemmer*

Tabel 4. 25 Hasil Analisis Sentimen *Poter Stemmer* Dataset 1

Index	text_porter	senti_porter_score	senti_porter_label
49995	thought movi right good job wasnt creativ orig...	1.500	Positif
49996	bad plot bad dialogu bad act idiot direct anno...	-1.750	Negatif
49997	cathol taught parochi elementari school nun ta...	-1.000	Negatif
49998	im go disagre previou comment side maltin one ...	-1.625	Negatif
49999	one expect star trek movi high art fan expect ...	2.125	Positif

Tabel 4. 26 Hasil Analisis Sentimen *Poter Stemmer* Dataset 2

Index	text_porter	senti_porter_score	senti_porter_label
14635	thank got differ flight chicago	0.000	Netral
14636	leav minut late flight warn commun minut late ...	0.375	Positif
14637	pleas bring american airlin	- 0.125	Negatif
14638	money chang flight dont answer phone suggest m...	0.500	Positif
14639	ppl need know mani seat next flight plz put us…	0.000	Netral

Tabel 4. 27 Hasil Analisis Sentimen *Poter Stemmer* Dataset 3

Index	text_porter	senti_porter_score	senti_porter_label
1599995	woke school best feel ever	0.875	Positif
1599996	thewdbcom cool hear old walt interview	0.250	Positif
1599997	readi mojo makeov ask detail	0.125	Positif
1599998	happi th birthday boo alll time tupac amaru sh...	0.000	Netral
1599999	happi	0.000	Netral

- *Snowball Stemmer*

Tabel 4. 28 Hasil Analisis Sentimen *Snowball Stemmer* Dataset 1

Index	text_snowball	senti_snowball_score	senti_snowball_label
49995	thought movi right good job wasnt creativ orig...	1.500	Positif
49996	bad plot bad dialogu bad act idiot direct anno...	- 1.750	Negatif
49997	cathol taught parochi elementari school nun ta...	- 1.000	Negatif
49998	im go disagre previous comment side maltin one...	- 2.250	Negatif
49999	one expect star trek movi high art fan expect ...	1.125	Positif

Tabel 4. 29 Hasil Analisis Sentimen *Snowball Stemmer* Dataset 2

Index	text_snowball	senti_snowball_score	senti_snowball_label
14635	thank got differ flight chicago	0.000	Netral
14636	leav minut late flight warn communic minut lat...	0.375	Positif
14637	pleas bring american airlin	- 0.125	Negatif
14638	money chang flight dont answer phone suggest m...	0.500	Positif
14639	ppl need know mani seat next flight plz put us...	0.000	Netral

Tabel 4. 30 Hasil Analisis Sentimen *Snowball Stemmer* Dataset 3

Index	text_snowball	senti_snowball_score	senti_snowball_label
1599995	woke school best feel ever	0.875	Positif
1599996	thewdbcom cool hear old walt interview	0.250	Positif
1599997	readi mojo makeov ask detail	0.125	Positif
1599998	happi th birthday boo alll time tupac amaru sh...	0.000	Netral
1599999	happi	0.000	Netral

- *Wordnet Lemmatizer*

Tabel 4. 31 Hasil Analisis Sentimen *Wordnet Lemmatizer* Dataset 1

Index	text_lemmatize	senti_lemmatize_score	senti_lemmatize_label
49995	thought movie right good job wasnt creative or...	2.125	Positif
49996	bad plot bad dialogue bad acting idiotic direc..	- 1.250	Negatif
49997	catholic taught parochial elementary school nu...	- 0.750	Negatif
49998	im going disagree previous comment side maltin...	- 5.000	Negatif
49999	one expects star trek movie high art fan expec...	1.625	Positif

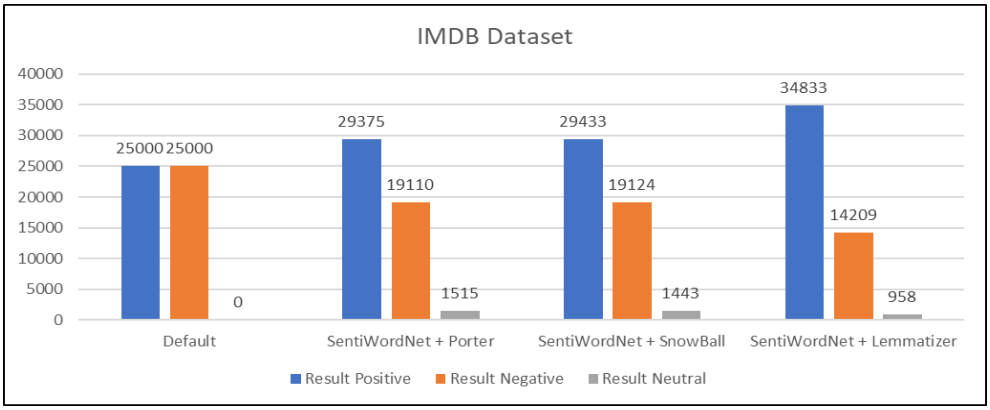
Tabel 4. 32 Hasil Analisis Sentimen *Wordnet Lemmatizer* Dataset 2

Index	text_lemmatize	senti_lemmatize_score	senti_lemmatize_label
14635	thank got different flight chicago	0.625	Positif
14636	leaving minute late flight warning communicati...	- 0.750	Negatif
14637	please bring american airline	0.000	Netral
14638	money change flight dont answer phone suggesti...	0.500	Positif
14639	ppl need know many seat next flight plz put u ...	0.250	Positif

Tabel 4. 33 Hasil Analisis Sentimen *Wordnet Lemmatizer* Dataset 3

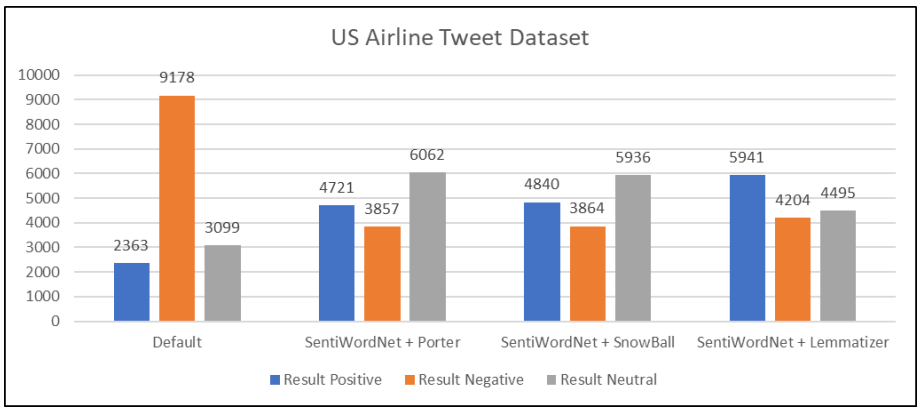
Index	text_lemmatize	senti_lemmatize_score	senti_lemmatize_label
1599995	woke school best feeling ever	0.750	Positif
1599996	thewdbcom cool hear old walt interview	0.250	Positif
1599997	ready mojo makeover ask detail	0.125	Positif
1599998	happy th birthday boo alll time tupac amaru sh...	0.875	Positif
1599999	happy	0.875	Positif

Dari proses analisis sentimen menggunakan lexion-based terkhusus *SentiWordNet* (SWN) *lexical resource* atau *dictionary* dihasilkan polarity dan label yang berbeda dari setiap teks *stemmer*. Berikut adalah hasil statistik analisis sentimen dari setiap teks *stemmer* pada dataset-dataset yang digunakan.



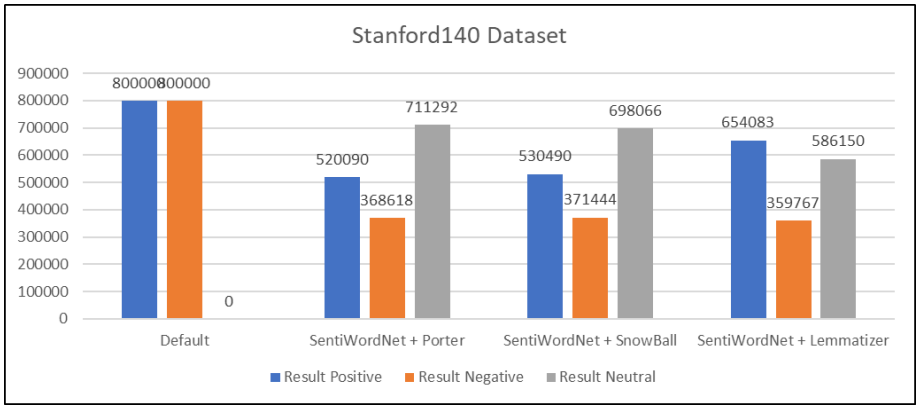
Gambar 4. 1 Statistik Hasil Analisis Sentimen Dataset 1

Dapat dilihat dari statistik diatas, analisis sentimen pada dataset 1 (IMDB Dataset) cenderung menghasilkan sentimen positif dibandingkan dengan sentimen negatif dan netral. Rata-rata sentimen positif semua *stemmer* adalah 31213,67 data, 17481 data sentimen negatif dan 1305,3 untuk sentimen netral.



Gambar 4. 2 Statistik Hasil Analisis Sentimen Dataset 2

Analisis sentimen pada dataset 2 (US Airline Tweet Dataset) berdasarkan statistik di atas cenderung menghasilkan sentimen netral dibandingkan dengan sentimen positif dan negatif. Dari 14640 data, rata-rata sentimen netral adalah sebanyak 5497,67 data. Sedangkan untuk sentimen positive dan negatif sebanyak 5167,3 dan 3975 data.



Gambar 4. 3 Statistik Hasil Analisis Sentimen Dataset 3

Pada dataset 3 (Stanford Sentiment140 Dataset) juga menghasilkan sentimen netral lebih banyak dibandingkan kedua sentimen lainnya. Sentimen netral yang dihasilkan dari semua stemmer adalah 665169,3 rata-rata. Sementara untuk sentiment positive dan negative rata-rata sebanyak 568221 dan 366609,67 data.

Terjadinya perbedaan hasil sentiment pada setiap dataset baik berdasarkan teks *stemmer* yang digunakan, dikarenakan beberapa faktor seperti berikut:

1. Dataset yang berbeda baik dari jumlah ataupun isi dari dataset sendiri.
2. *Stemming* juga berpengaruh pada hasil analisis sentimen karena pada proses ini kata-kata dapat berubah menjadi bentuk akar atau dasar yang mungkin tidak sepenuhnya mempresentasikan makna asli kata.
3. Terakhir adalah pada SWN *lexical resource* sendiri. Dimana SWN tidak mencakup semua kata yang kehilangan informasi kontekstual berdasarkan hasil *stemming*.

4.4. *Feature Extraction*

Feature extraction digunakan untuk mengubah data menjadi format yang kemudian dapat digunakan oleh algoritma *machine learning*. Proses ini akan merubah teks disiapkan untuk data tabular. Seperti yang telah dipaparkan pada bagian 3.4.5, jenis *feature extraction* yang digunakan pada penelitian ini diantaranya adalah *N-grams,* dan *Bag-of-Words* (BoW). *Feature extraction* diimplemtasi pada variabel X, dimana variabel X berisi teks hasil dari semua *stemmer* yang digunakan. Pada bagian 4.4.1 dan 4.4.2 akan dipaparkan hasil dari implementasi dai *feature extraction*.

4.4.1. N-gram

Implementasi N-gram *feature extraction* pada penelitian menggunakan *library HashingVectorizer* dari *sklearn*. ngram_range yang digunakan adalah (1,3). Penggunaan *HashingVectorizer* didasarkan pada efesiensi dan skalabilitas, *HashingVectorizer* dapat menangani data yang besar dengan mudah karena hanya membutuhkan sedikit memori. Ngram_range (1, 3) akan menghasilkan N-gram dengan panjang 1, 2, dan 3 kata. Ini memungkinkan kombinasi unigram, bigram, dan trigram dapat meningkatkan akurasi model. Tabel 4. 34 – 4. 36 menampilkan distribusi frekuensi dari N-grams pada hasil *stemmer* di setiap dataset.

Tabel 4. 34 Distribusi Frekuensi dari N-grams Dataset 1

IMDb	Range of N-gram frequencies		Most frequent N-gram count	Least frequent N-gram count
	from	to		
Porter + N-Gram	-5.036	48.248	1048076	0
SnowBall + N-Gram	-5.035	48.240	1048072	0
Lemmatizer + N-Gram	-5.067	49.562	1048026	0

Tabel 4. 35 Distribusi Frekuensi dari N-grams Dataset 2

Twitter US Airline Sentiment	Range of N-gram frequencies		Most frequent N-gram count	Least frequent N-gram count
	from	to		
Porter + N-Gram	-9.346	3.170	1048102	0
SnowBall + N-Gram	-9.363	31.702	1048110	0
Lemmatizer + N-Gram	-8.933	31.915	1047814	0

Tabel 4. 36 Distribusi Frekuensi dari N-grams Dataset 3

Sentiment140	Range of N-gram frequencies		Most frequent N-gram count	Least frequent N-gram count
	from	to		
Porter + N-Gram	-3.063	3.890	1043869	0
SnowBall + N-Gram	-3.062	38.908	1043462	0
Lemmatizer + N-Gram	-2.383	38.960	1042930	0

Tabel distribusi frekuensi dari N-grams menunjukkan hasil analisis frekuensi N-Gram berdasarkan tiga dataset dan tiga hasil *stemming* yang berbeda. Adapun faktor-faktor yang mempengaruhi hasil distribusi frekuensi dari N-gram adalah sebagai berikut.

1. *Range* of N-gram *Frequencies* dari setiap dataset memiliki hasil yang berbeda, dikarenakan setiap hasil *stemming* memiliki aturan tersendiri dalam mengolah kata sehingga menghasilkan frekuensi kemunculan N-gram berbeda.
2. Kualitas dan jumlah dataset juga mempengaruhi hasil dari *Range* of N-gram. Dapat diperhatikan bahwasanya untuk dataset Twitter US Airline Sentiment memiliki frekuensi yang lebih kecil disbanding dengan dataset IMDb. Dan dataset Sentiment140 memiliki rentang frekuensi yang paling luas diantara semua dataset. Karena dari jumlah dataset Sentiment140 adalah yang paling besar.

3. *Most Frequent* N-gram *Count* juga dipengaruhi oleh faktor dataset, dimana dataset yang lebih besar cenderung memiliki N-gram yang lebih sering muncul

4.4.2. Bag-of-Words (Bow)

Implementasi BoW *feature extraction* pada juga menggunakan *library HashingVectorizer* dari sklearn tanpa menggunakan ngram_range. Sama halnya dengan N-Gram penggunaan *HashingVectorizer* didasarkan pada efesiensi dan skalabilitas, *HashingVectorizer* dapat menangani data yang besar dengan mudah karena hanya membutuhkan sedikit memori. Sebaliknya pada BoW tidak menggunakan ngram_range karena model ini tidak mempertimbangkan urutan kata. Dan pada BoW hanya membutuhkan informasi tentang kata-kata individual (unigram), sehingga tidak perlu menggunakan N-gram dengan panjang lebih dari 1. Tabel 4. 37 – 4. 39 menampilkan distribusi frekuensi dari Words (BoW) pada hasil *stemmer* di setiap dataset.

Tabel 4. 37 Distribusi Frekuensi dari Words (BoW) Dataset 1

IMDb	Range of word frequencies		Most frequent word count	Least frequent word count
	from	to		
Porter + BoW	-7.545	71.307	71.307	-7.545
SnowBall + BoW	-7.542	7.128	7.128	-7.542
Lemmatizer + BoW	-76.366	73.714	73.714	-76.366

Tabel 4. 38 Distribusi Frekuensi dari Words (BoW) Dataset 2

Twitter US Airline Sentiment	Range of word frequencies		Most frequent word count	Least frequent word count
	from	to		
Porter + BoW	-14.651	50.471	50.471	-14.651
SnowBall + BoW	-1.467	50.471	50.471	-1.467
Lemmatizer + BoW	-14.066	4.491	4.491	-14.066

Tabel 4. 39 Distribusi Frekuensi dari Words (BoW) Dataset 3

Sentiment140	Range of word frequencies		Most frequent word count	Least frequent word count
	from	to		
Porter + BoW	-4.770	6.081	6.081	-4.770
SnowBall + BoW	-4.769	60.815	60.815	-4.769
Lemmatizer + BoW	-36.962	60.952	60.952	-36.962

Tabel distribusi frekuensi dari words (kata) menunjukkan hasil analisis frekuensi data berdasarkan tiga dataset dan tiga hasil *stemming* yang berbeda. Adapun faktor-faktor yang mempengaruhi hasil distribusi frekuensi kata adalah sebagai berikut.

1. Sama halnya dengan *Range* of N-gram *Frequencies, Range of Words Frequencies* dari setiap dataset memiliki hasil yang berbeda karena setiap hasil *stemming* memiliki pengaruh terhadap kemunculan kata. Dataset yang lebih besar dan lebih beragam cenderung memiliki rentang frekuensi yang lebih luas.
2. *Most Frequent Word Count* dipengaruhi oleh faktor sering munculnya kata umum atau frasa yang banyak digunakan. Serta ukuran dataset mempengaruhi kata-kata umum yang sering muncul
3. *Least Frequent Word Count* dipengaruhi oleh Noise atau Kesalahan. Dimana beberapa kata mungkin tidak muncul sama sekali karena *noise* atau kesalahan dalam data atau proses pengolahan data.

Maka dapat disimpulkan bahwa distribusi frekuensi dari n-grams distribusi frekuensi dari words pada bow, dipengaruhi oleh kombinasi metode pengolahan teks, kualitas dataset, dan parameter yang digunakan dalam analisis untuk N-gram.

4.5. Implementasi Model (Support Vector Machine)

Pada tahap ini ada tahapan-tahapan sebelumnya yang perlu dilakukan, dengan maksud untuk menghasilkan akurasi yang tinggi pada model SVM. Tahapan-tahapan yang perlu dilakukan sebelumnya adalah *handling imbalance* menggunakan *library smote* dari *imblearn* dan *splitting* data.

Handling imbalance dilakukan karena data teks sering kali memiliki distribusi data yang tidak seimbang. Tahap ini menggunakan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) yang termasuk dalam kategori *oversampling*. Dari tahap *handling imbalance* dihasilkan nilai *counter* (jumlah) yang berbeda dari *counter* utama pada variabel y yakni label sentimen. Berikut adalah statistik hasil *counter* dari stiap dataset berdasarkan *stemmer*.

Tabel 4. 40 Statistik *Handling Imbalance Porter Stemmer* Dataset 1

Dataset 1	Positif	Negatif	Netral
Porter	29375	19110	1515
Porter + Smote	29375	29375	29375

Tabel 4. 41 Statistik *Handling Imbalance SnowBall Stemmer* Dataset 1

Dataset 1	Positif	Negatif	Netral
Snowball	29433	19124	1443
Snowball + Smote	29433	29433	29433

Tabel 4. 42 Statistik *Handling Imbalance Wordnet Lemmatizer* Dataset 1

Dataset 1	Positif	Negatif	Netral
Lemmatizer	34833	14209	958
Lemmatizer + Smote	34833	34833	34833

Tabel 4. 43 Statistik *Handling Imbalance Porter Stemmer* Dataset 2

Dataset 2	Positif	Negatif	Netral
Porter	4721	3857	6062
Porter + Smote	6062	6062	6062

Tabel 4. 44 Statistik *Handling Imbalance SnowBall Stemmer* Dataset 2

Dataset 2	Positif	Negatif	Netral
Snowball	4840	3864	5936
Snowball + Smote	5936	5936	5936

Tabel 4. 45 Statistik *Handling Imbalance Wordnet Lemmatizer* Dataset 2

Dataset 2	Positif	Negatif	Netral
Lemmatizer	5941	4204	4495
Lemmatizer + Smote	5941	5941	5941

Tabel 4. 46 Statistik *Handling Imbalance Porter Stemmer* Dataset 3

Dataset 3	Positif	Negatif	Netral
Porter	520090	368618	711292
Porter + Smote	711292	711292	711292

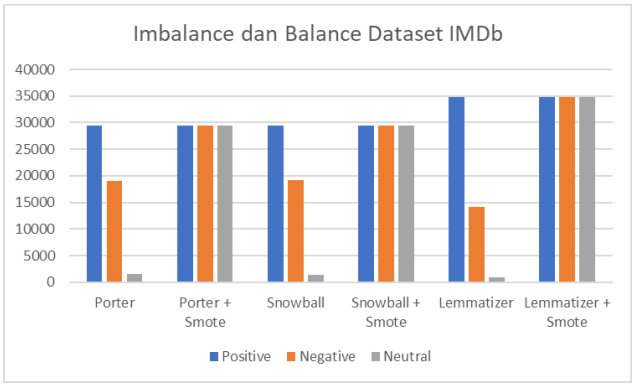
Tabel 4. 47 Statistik *Handling Imbalance SnowBall Stemmer* Dataset 3

Dataset 3	Positif	Negatif	Netral
Snowball	530490	371444	698066
Snowball + Smote	698066	698066	698066

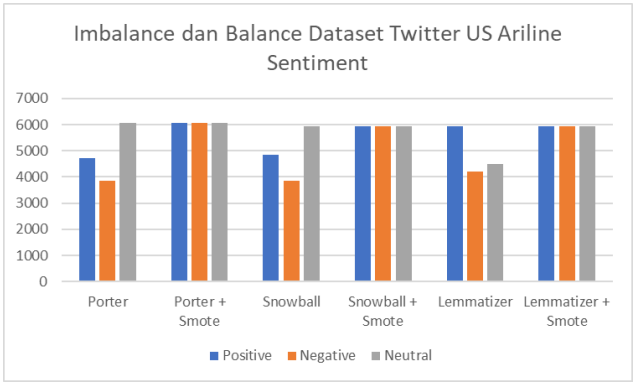
Tabel 4. 48 Statistik *Handling Imbalance Wordnet Lemmatizer* Dataset 3

Dataset 3	Positif	Negatif	Netral
Lemmatizer	654083	359767	586150
Lemmatizer + Smote	654083	654083	654083

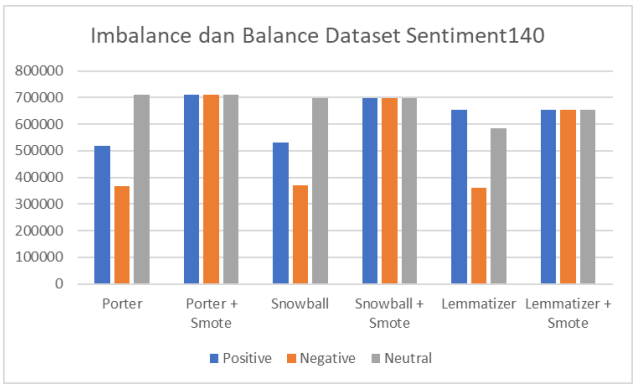
Untuk mempermudah dalam memahami tabel yang sudah tersaji berikut adalah grafik perbandingan antara data yang belum balance dan yang sudah dilakukan handling imbalance berdasarkan *stemmer* dalam tiga dataset pada gambar 4.4 – 4.6.



Gambar 4. 4 Grafik *Imbalance* dan *Balance* Dataset 1



Gambar 4. 5 Grafik *Imbalance* dan *Balance* Dataset 2



Gambar 4. 6 Grafik *Imbalance* dan *Balance* Dataset 3

Berdasarkan hasil dari *handling imbalance* menggunakan SMOTE dapat dilihat bahwasanya setiap data pada *class* mengikuti data mayoritas. Ini dikarenakan cara kerja SMOTE yang memilih secara acak sebuah sample *class* minoritas yang kemudian mencari tetangga terdekat dari sampel. Setelahnya smote membuat data sintesis antara sample asli dan salah satu tetannga terdekta yang dibilih acak. Data sintetis ini dibuat dengan menggabungkan fitur-fitur dari sampel asli dengan perbedaan fitur antara sampel asli dan tetangga yang dikalikan dengan bilangan acak antara 0 dan 1. Proses ini dilakukan secara berulang hingga jumlah data di kelas minoritas sama dengan jumlah data di kelas mayoritas. Dari proses handling imbalance ini akan dihasilkan dataset yang lebih seimbang.

Tahap berikutnya adalah *splitting* data. Pembagian ini penting dalam implementasi *machine learning*. Seperti mencegah *overfitting* dimana *training* set digunakan untuk melatih model, sedangkan *testing* set digunakan untuk mengevaluasi performa model. Pembagian ini membantu memastikan bahwa model tidak hanya mempelajari detail *training* set (*overfitting*) dan dapat menggeneralisasi dengan baik ke data baru (*testing* set). Pada proses *splitting* data peneliti mengunggunakan perbandingan 20:80 untuk membagi *test* set dan *training* set. Berikut merupakan statistik dari *splitting* data pada dataset berdasarkan kombinasi dari *stemmer* dan *feature extraction* yang digunakan.

Tabel 4. 49 Statistik Data Split Dataset 1

Combination Dataset 1	Training			Testing		
	Positive	Negative	Neutral	Positive	Negative	Neutral
Porter + N-Gram	23537	23562	23401	5838	5813	5974
Porter + BoW	23546	23484	23470	5829	5891	5905
SnowBall + N-Gram	23582	23547	23510	5851	5886	5923
SnowBall + BoW	23529	23492	23618	5904	5941	5815
Lemmatizer + N-Gram	27830	27888	27881	7003	6945	6952
Lemmatizer + BoW	27802	27896	27901	7031	6937	6932

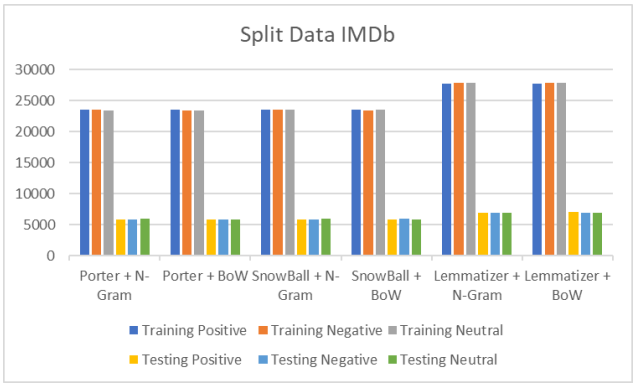
Tabel 4. 50 Statistik Data Split Dataset 2

Combination Dataset 2	Training			Testing		
	Positive	Negative	Neutral	Positive	Negative	Neutral
Porter + N-Gram	4842	4834	4872	1220	1228	1190
Porter + BoW	4868	4854	4826	1194	1208	1236
SnowBall + N-Gram	4789	4745	4712	1147	1191	1224
SnowBall + BoW	4769	4742	4735	1167	1194	1201
Lemmatizer + N-Gram	4767	4776	4715	1174	1165	1226
Lemmatizer + BoW	4696	4769	4793	1245	1172	1148

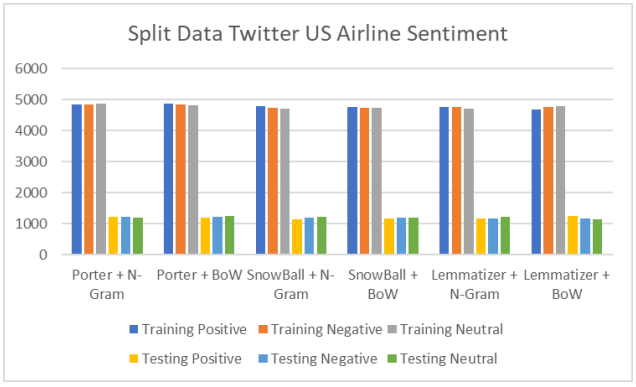
Tabel 4. 51 Statistik Data Split Dataset 3

Combination Dataset 3	Training			Testing		
	Positive	Negative	Neutral	Positive	Negative	Neutral
Porter + N-Gram	568972	569786	568342	142320	141506	142950
Porter + BoW	569083	568998	569019	142209	142294	142273
SnowBall + N-Gram	558366	559292	557700	139700	138774	140366
SnowBall + BoW	558596	558115	558647	139470	139951	139419
Lemmatizer + N-Gram	523378	523685	522736	130705	130398	131347
Lemmatizer + BoW	523279	523318	523202	130804	130765	130881

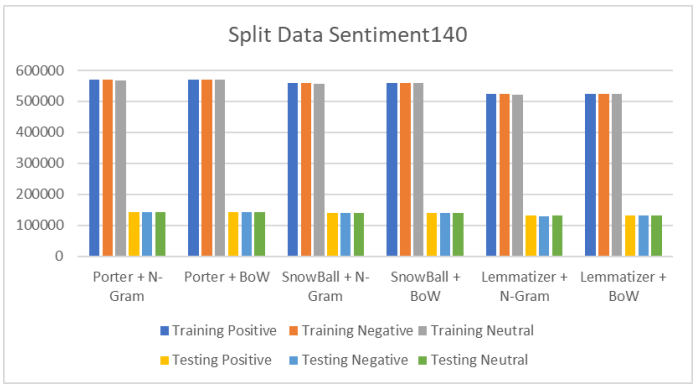
Pada gambar 4.7 – 4.9 ditampilkan grafik perbandingan antara *training* dan *testing* data berdasarkan skenario yang digunakan pada 3 dataset.



Gambar 4. 7 Grafik Split Data Dataset 1



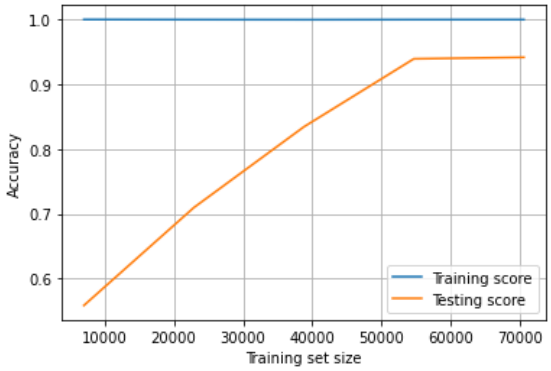
Gambar 4. 8 Grafik Split Data Dataset 2



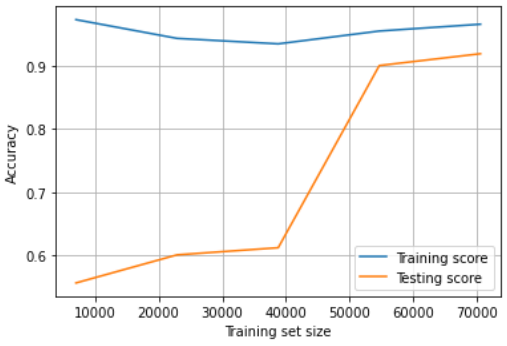
Gambar 4. 9 Grafik Split Data Dataset 3

Setelah tahapan dilakukan maka tahap selanjutkan adalah mendefinisikan model klasifikasi yang digunakan yakni *LinearSVC*. Kemudian model dilatih pada data *training* dan label sentimen. Selanjutnya menggunakan model yang sudah dilatih untuk memprediksi sentimen pada data *testing*. Berikutnya dilakukan analisa kinerja model terhadap data latih.

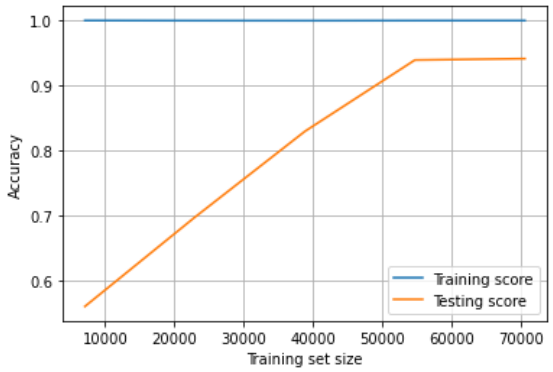
Analisa kinerja model penting dilakukan untuk mengetahui apakah model berpotensi mengalami *overfitting* atau tidak. Analisa kinerja model pada penelitian menggunakan learning curve dari library sklearn. Langkah ini dilakukan berdasarkan skenario gabungan antara stemmer dan feature extraction pada setiap dataset.



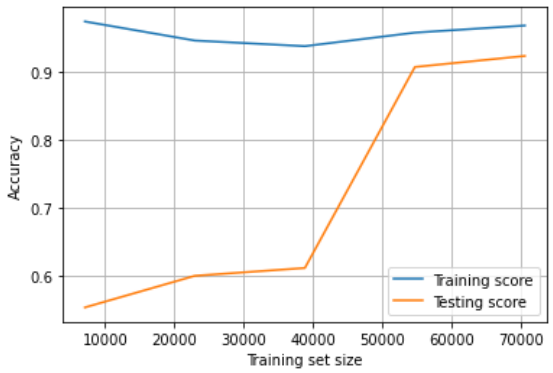
Gambar 4. 10 *Learning Curve Analysist Porter Stemmer* dan *N-Gram* Dataset 1



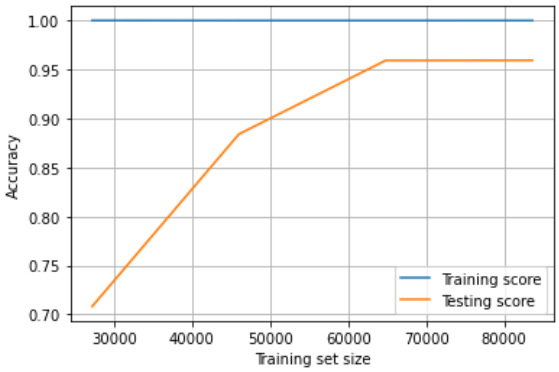
Gambar 4. 11 *Learning Curve Analysist Porter Stemmer* dan *BoW* Dataset 1



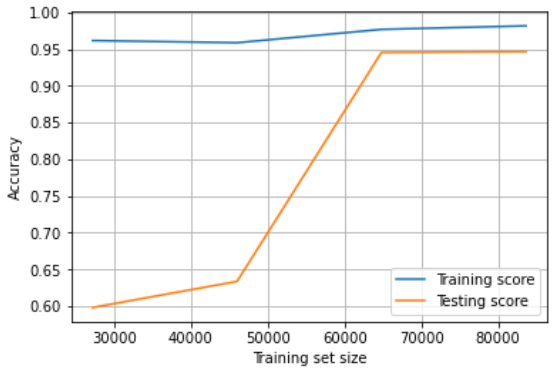
Gambar 4. 12 *Learning Curve Analysist SnowBall Stemmer* dan *N-Gram* Dataset 1



Gambar 4. 13 *Learning Curve Analysist SnowBall Stemmer* dan *BoW* Dataset 1



Gambar 4. 14 *Learning Curve Analysist Wordnet Lemmatizer* dan *N-Gram* Dataset 1



Gambar 4. 15 *Learning Curve Analysist Wordnet Lemmatizer* dan *BoW* Dataset 1

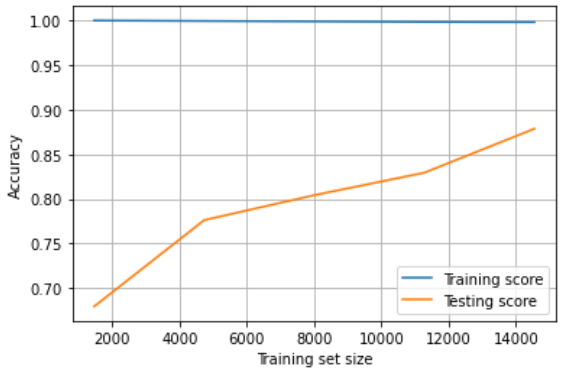
Gambar 4.10 – 4.15, merupakan kurva analisa kinerja model pada dataset IMDb. Trend pada traning set semuanya menurun sedangkan pada *testing* set terjadi penuruan pada kombinasi *Lemmatizer* dan N-Gram serta *Lemmatizer* dan Bow. Pada Tabel 4. 52 ditampilkan statistik dari training dan testing set performance.

Tabel 4. 52 Statistik *Training* dan *Testing Set Performance* Dataset 1

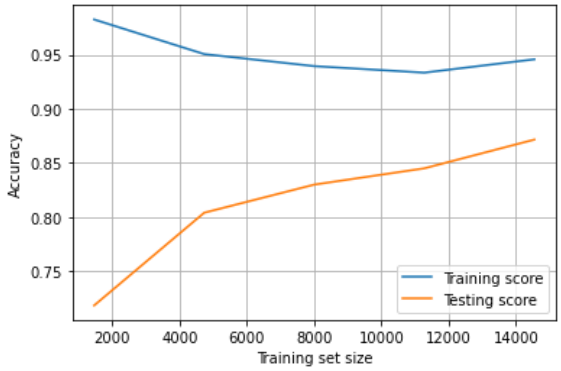
IMDb	Training Set Performance			Testing Set Performance			Gap Training and Testing			EoO
	Initial accuracy	Final accuracy	Trend	Initial accuracy	Final accuracy	Trend	Initial gap	Final gap	Trend	
Porter + N-Gram	1,0	0,9999	dec	0,5585	0,9416	inc	0,4415	0,0582	Dec	no
Porter + BoW	0,9738	0,9665	dec	0,5561	0,9197	inc	0,4177	0,0468	Dec	no
SnowBall + N-Gram	1,0	0,9999	dec	0,5597	0,9413	inc	0,4403	0,0586	Dec	no
SnowBall + BoW	0,9736	0,9676	dec	0,5543	0,9230	inc	0,4192	0,0446	Dec	no
Lemmatizer + N-Gram	nan	0,9999	dec	nan	0,9593	dec	nan	0,0406	Dec	no
Lemmatizer + BoW	nan	0,9818	dec	nan	0,9465	dec	nan	0,0353	Dec	no

Pada tabel 4.52 dapat diperhatikan *initial accuracy* pada *training* set sangat tinggi (1.0 atau mendekati 1.0) dalam *Porter* dan *Snowball* stemmer dengan N-Gram dan BOW, tetapi akurasi akhir menurun. Ini menunjukkan bahwa model ini

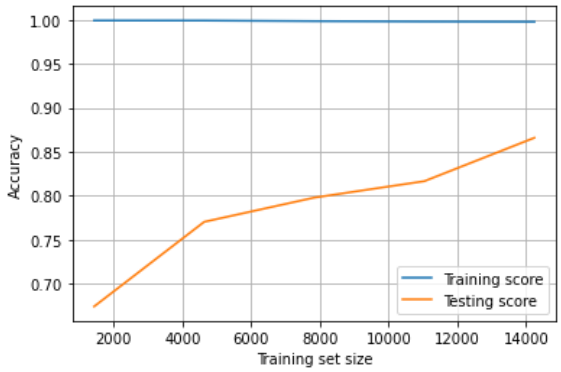
overfitting pada data latihan. Penurunan akurasi pada akhirnya dapat disebabkan oleh kelebihan penyesuaian terhadap data pelatihan. Sedangkan pada *Lemmatizer* didapatkan nilai ‘nan’ pada *initial accuracy*, yang mungkin disebabkan oleh kesalahan atau masalah dalam pengolahan data. Di sisi lain pada tren “*Testing Set Performance*” semua metode kecuali pada metode *Lemmatizer* dan N-Gram serta BoW menunjukkan peningkatan meskipun ada penurunan dalam performa *training* set. Ini menunjukkan bahwa meskipun akurasi pada *training* set menurun, model lebih baik dalam menggeneralisasi pada *testing* set.



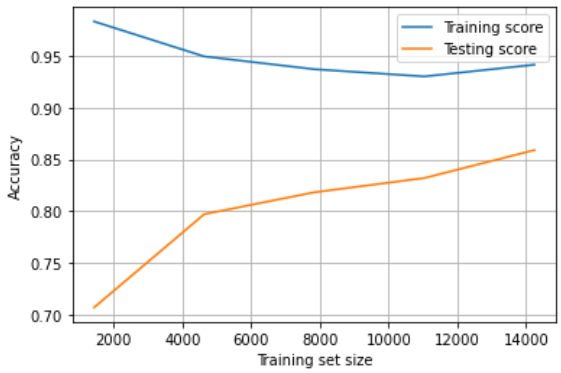
Gambar 4. 16 *Learning Curve Analysist Porter Stemmer* dan *N-Gram* Dataset 2



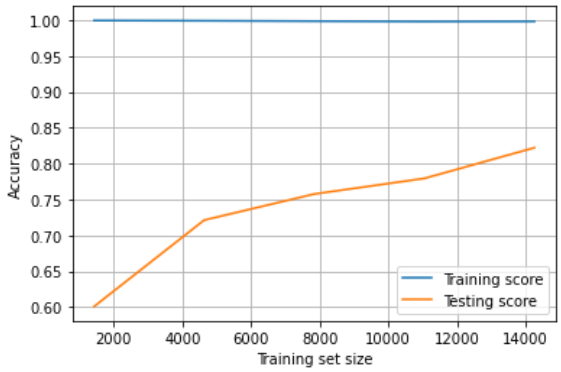
Gambar 4. 17 *Learning Curve Analysist Porter Stemmer* dan *BoW* Dataset 2



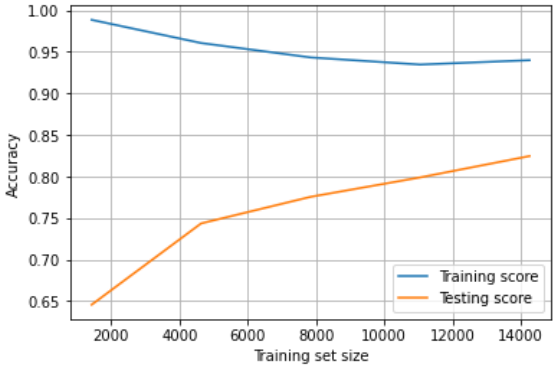
Gambar 4. 18 *Learning Curve Analysist SnowBall Stemmer* dan *N-Gram* Dataset 2



Gambar 4. 19 *Learning Curve Analysist SnowBall Stemmer* dan *BoW* Dataset 2



Gambar 4. 20 *Learning Curve Analysist Wordnet Lemmatizer* dan *N-Gram* Dataset 2



Gambar 4. 21 *Learning Curve Analysist Wordnet Lemmatizer* dan *BoW* Dataset 2

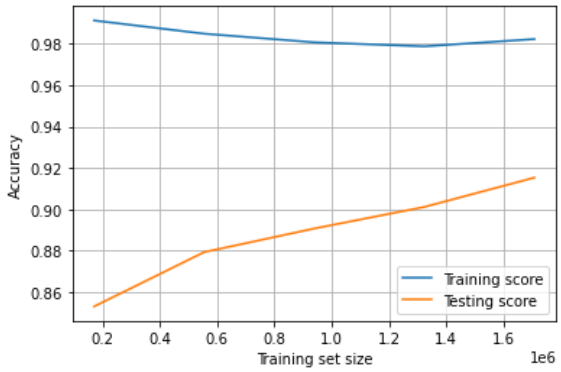
Gambar 4.16 – 4.12, merupakan kurva analisa kinerja model pada dataset Twitter US Airline Sentiment. Trend pada *training* set semuanya menurun sedangkan pada *testing* set semuanya naik. Dan trend pada *gap training* dan *testing* set juga menurun seperti yang bisa dilihat pada Tabel 4. 53.

Tabel 4. 53 Statistik *Training* dan *Testing Set Peformance* Dataset 2

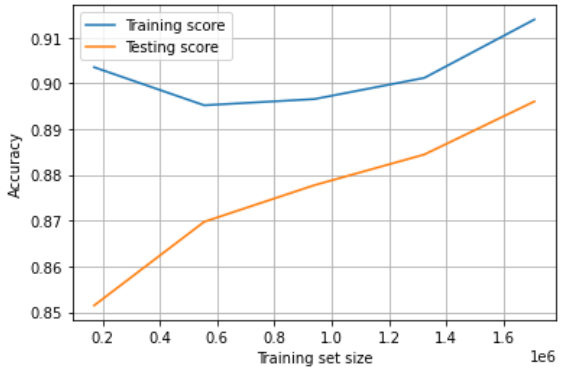
Twitter US Airline Sentiment	Training Set Performance			Testing Set Performance			Gap Training and Testing			EoO
	Initial accuracy	Final accuracy	Trend	Initial accuracy	Final accuracy	Trend	Initial gap	Final gap	Trend	
Porter + N-Gram	1,0	0,9981	dec	0,6794	0,8786	inc	0,3206	0,1195	dec	yes
Porter + BoW	0,9827	0,9458	dec	0,7180	0,8715	inc	0,2647	0,0743	dec	no
SnowBall + N-Gram	0,9994	0,9980	dec	0,6739	0,8659	inc	0,3255	0,1321	dec	yes
SnowBall + BoW	0,9831	0,9414	dec	0,7073	0,8591	inc	0,2759	0,0824	dec	no
Lemmatizer + N-Gram	0,9996	0,9984	dec	0,6008	0,8221	inc	0,3988	0,1762	dec	yes
Lemmatizer + BoW	0,9884	0,9398	dec	0,6456	0,8244	inc	0,3427	0,1153	dec	yes

Pada tabel 4.53 gabungan antara *Porter*, *Snowball*, *Lemmatizer* dan N-gram dan *Lemmatizer* dan BoW terdapat bukti kemungkinan *overfitting* karena selisih yang signifikan antara kinerja pelatihan dan pengujian. Pada Trend "*Training Set*

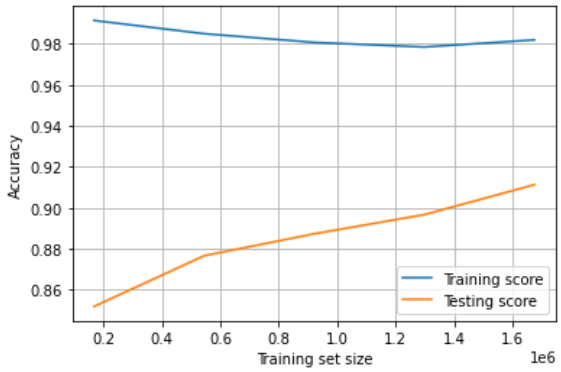
Performance" semua metode menunjukkan peningkatan ini menunjukkan bahwa meskipun akurasi pada *training* set menurun, model lebih baik dalam menggeneralisasi pada *testing* set.



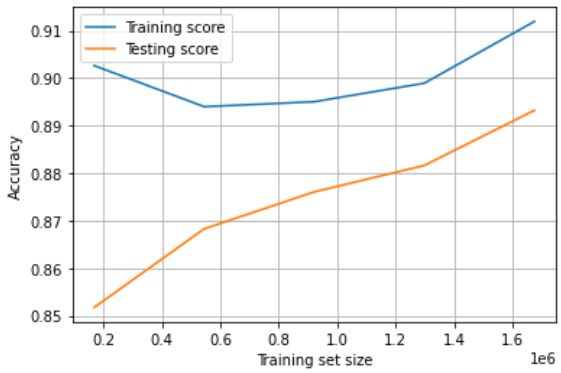
Gambar 4. 22 *Learning Curve Analysist Porter Stemmer* dan *N-Gram* Dataset 3



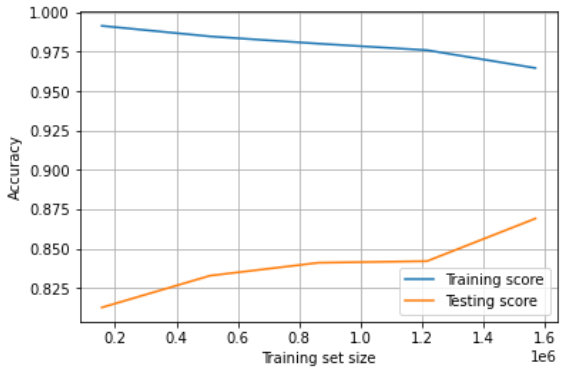
Gambar 4. 23 *Learning Curve Analysist Porter Stemmer* dan *BoW* Dataset 3



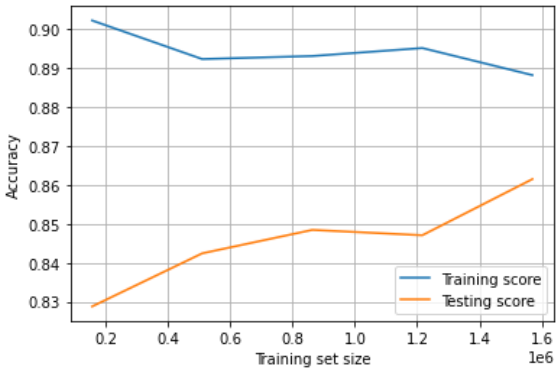
Gambar 4. 24 *Learning Curve Analysist SnowBall Stemmer* dan *N-Gram* Dataset 3



Gambar 4. 25 *Learning Curve Analysist SnowBall Stemmer* dan *BoW* Dataset 3



Gambar 4. 26 *Learning Curve Analysist Wordnet Lemmatizer* dan *N-Gram* Dataset 3



Gambar 4. 27 *Learning Curve Analysist Wordnet Lemmatizer* dan *BoW* Dataset 3

Gambar 4.22 – 4.27, merupakan kurva analisa kinerja model pada dataset Sentiment140. Ada kenaikan pada *training* set yaknik pada kombinasi *Porter* dan BoW serta *SnowBall* dan BoW selain itu trend menurun. Pada *testing* set semua trend naik. Dan trend pada gap *training* dan *testing* set juga menurun seperti yang bisa dilihat pada Tabel 4. 54.

Tabel 4. 54 Statistik *Training* dan *Testing Set Peformance* Dataset 3

Sentiment140	Training Set Performance			Testing Set Performance			Gap Training and Testing			EoO
	Initial accuracy	Final accuracy	Trend	Initial accuracy	Final accuracy	Trend	Initial gap	Final gap	Trend	
Porter + N-Gram	0,9914	0,9824	dec	0,8530	0,9153	inc	0,1383	0,0670	dec	no
Porter + BoW	0,9035	0,9140	inc	0,8515	0,8960	inc	0,0521	0,0179	dec	no
SnowBall + N-Gram	0,9914	0,9819	dec	0,8519	0,9112	inc	0,1395	0,0706	dec	no
SnowBall + BoW	0,9026	0,9119	inc	0,8517	0,8932	inc	0,0509	0,0187	dec	no
Lemmatizer + N-Gram	0,9914	0,9647	dec	0,8127	0,8692	inc	0,1787	0,0955	dec	no
Lemmatizer + BoW	0,9021	0,8881	dec	0,8288	0,8614	inc	0,0733	0,0267	dec	no

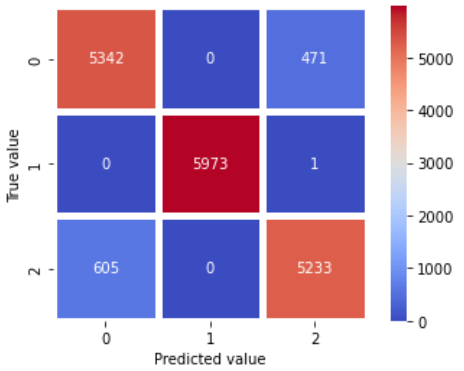
Pada tabel 4.54 terjadi penurunan tren di *training* set pada gabungan *Porter* dan N-Gram, *SnowBall* dan, N-Gram *Lemmatizer* dan N-Gram serta *Lemmatizer*

dan N-Gram. Seperti halnya pada dua pengujian sebelumnya pada Trend "*Training Set Performance*" semua metode menunjukkan peningkatan ini menunjukkan bahwa meskipun akurasi pada *training* set menurun, model lebih baik dalam menggeneralisasi pada *testing* set. Secara umum penurunan tren pada *training* set dapat disebabkan oleh kompleksitas model, sementara peningkatan pada *testing* set dimungkinan karena model yang lebih umum dan efektif dalam menggeneralisasi fitur-fiturnya.

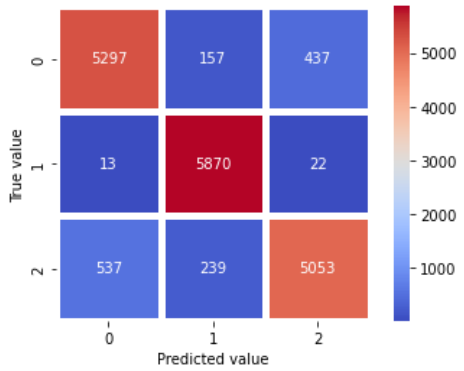
4.6. *Model Evaluation* (*Confusion Matrix*)

Setelah model berhasil diimplementasi langkah terakhir dari penelitian ini adalah mengevaluasi model. Evaluasi model pada penelitian ini menggunakan *confusion matrix* yang memberikan empat indikator yakni *accuracy, precision, recall* dan *F1-Score* seperti yang dipaparkan di bagian 3.4.7.

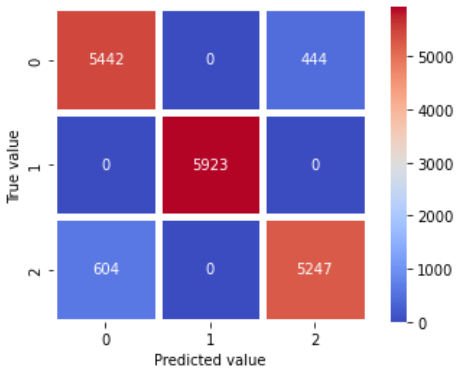
Evaluasi model SVM khususnya *LinearSVC* dilakukan pada semua skenario gambungan antara *stemmer* dan *feature extraction* dan diimplementasi pada tiga dataset yang berbeda. Pada gambar 4.28 – 4.33 adalah *confusion matrix* setiap skenario pada dataset 1, *confusion matrix* stiap skenario pada dataset 2 ditampilkan pada gambar 4.34 – 4.39, dan pada gambar 4.40 – 4.45 ditampilkan *confusion matrix* skenario pada dataset 3.



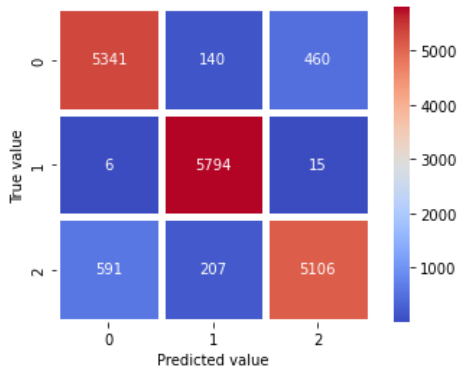
Gambar 4. 28 CM *Porter Stemmer* dan *N-Gram* Dataset 1



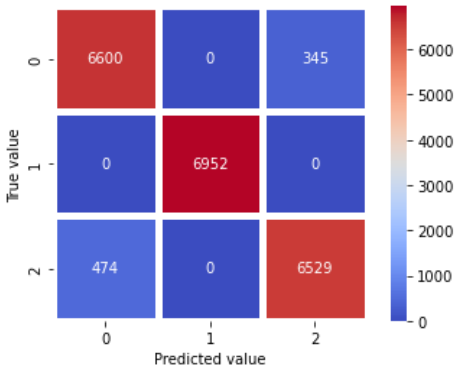
Gambar 4. 29 CM *Porter Stemmer* dan *BoW* Dataset 1



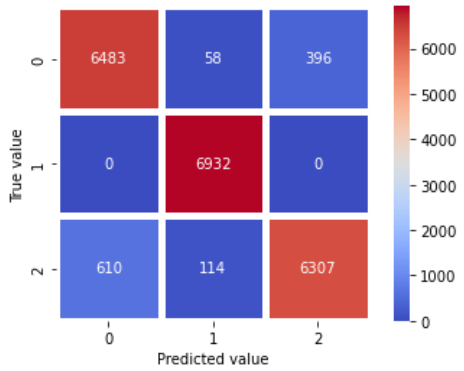
Gambar 4. 30 CM *Snowball Stemmer* dan *N-Gram* Dataset 1



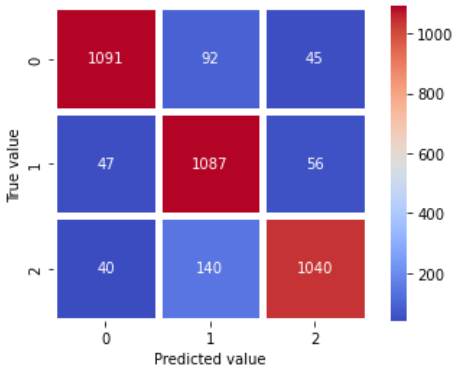
Gambar 4. 31 CM *Snowball Stemmer* dan *BoW* Dataset 1



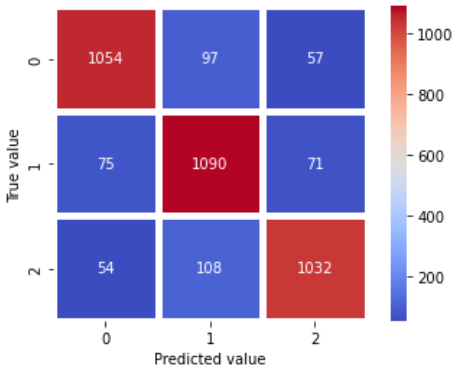
Gambar 4. 32 CM *WordNet Lemmatizer* dan *N-Gram* Dataset 1



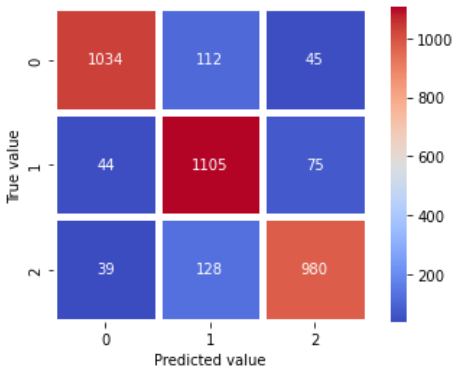
Gambar 4. 33 CM *WordNet Lemmatizer* dan *BoW* Dataset 1



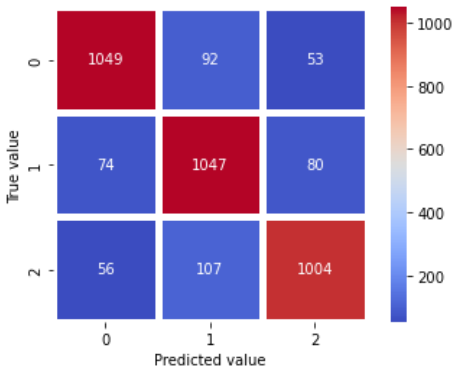
Gambar 4. 34 CM *Porter Stemmer* dan *N-Gram* Dataset 2



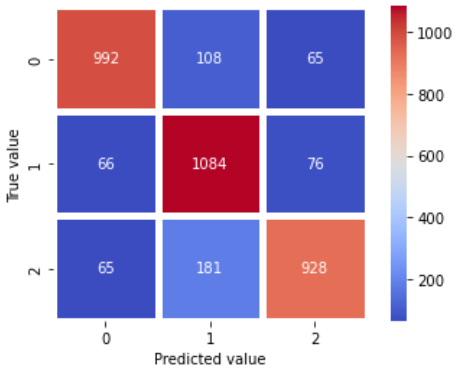
Gambar 4. 35 CM *Porter Stemmer* dan *BoW* Dataset 2



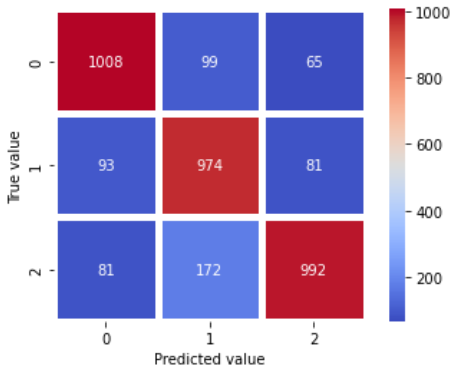
Gambar 4. 36 CM *Snowball Stemmer* dan *N-Gram* Dataset 2



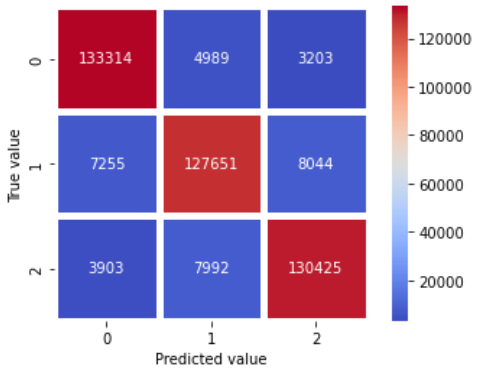
Gambar 4. 37 CM *Snowball Stemmer* dan *BoW* Dataset 2



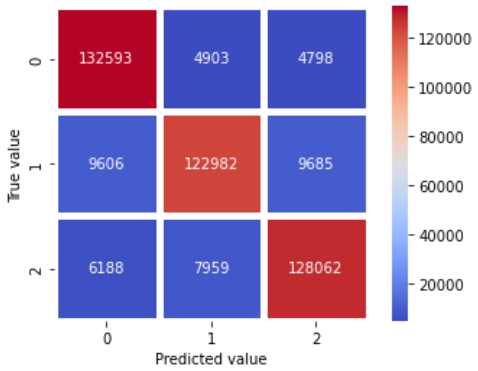
Gambar 4. 38 CM *WordNet Lemmatizer* dan *N-Gram* Dataset 2



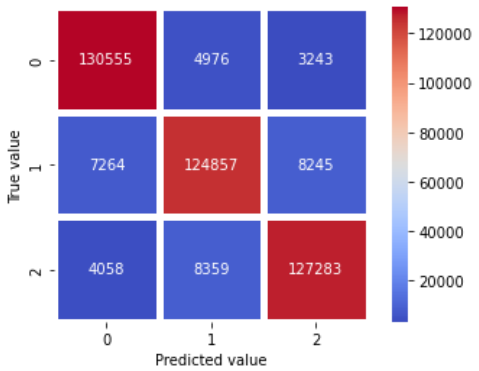
Gambar 4. 39 CM *WordNet Lemmatizer* dan *BoW* Dataset 2



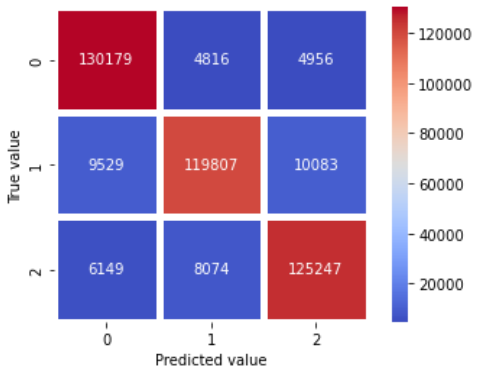
Gambar 4. 40 CM *Porter Stemmer* dan *N-Gram* Dataset 3



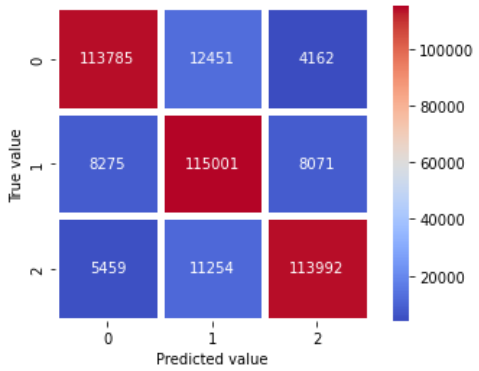
Gambar 4. 41 CM *Porter Stemmer* dan *BoW* Dataset 3



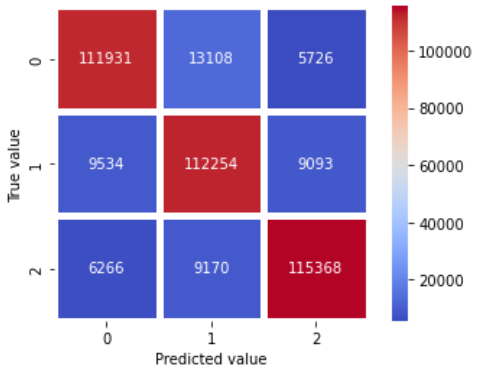
Gambar 4. 42 CM *Snowball Stemmer* dan *N-Gram* Dataset 3



Gambar 4. 43 CM *Snowball Stemmer* dan *BoW* Dataset 3



Gambar 4. 44 CM *WordNet Lemmatizer* dan *N-Gram* Dataset 3

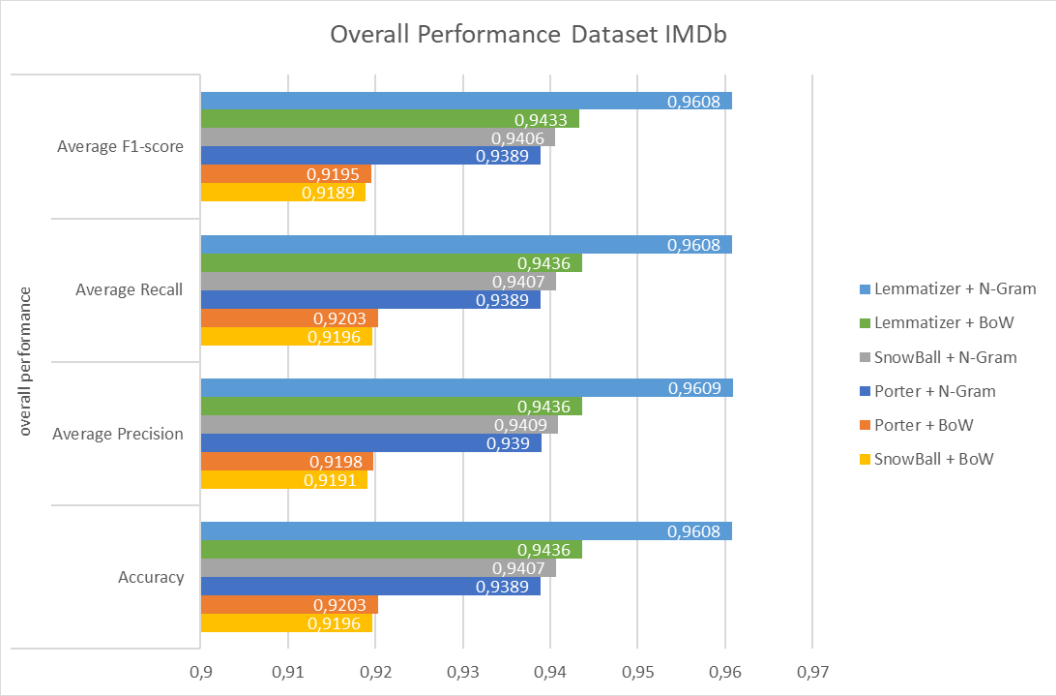


Gambar 4. 45 CM *WordNet Lemmatizer* dan *BoW* Dataset 3

Rangkuman hasil dari evaluasi model berdasarkan dataset ditampilkan pada Tabel 4. 55, Tabel 4. 56 dan Tabel 4. 57.

Tabel 4. 55 Hasil Evaluasi Model LiniearSVC Dataset 1

IMDB	Overall Performance			
	Accuracy	Average Precision	Average Recall	Average F1-score
Porter + N-Gram	0,9389	0,9390	0,9389	0,9389
Porter + BoW	0,9203	0,9198	0,9203	0,9195
SnowBall + N-Gram	0,9407	0,9409	0,9407	0,9406
SnowBall + BoW	0,9196	0,9191	0,9196	0,9189
Lemmatizer + N-Gram	0,9608	0,9609	0,9608	0,9608
Lemmatizer + BoW	0,9436	0,9436	0,9436	0,9433



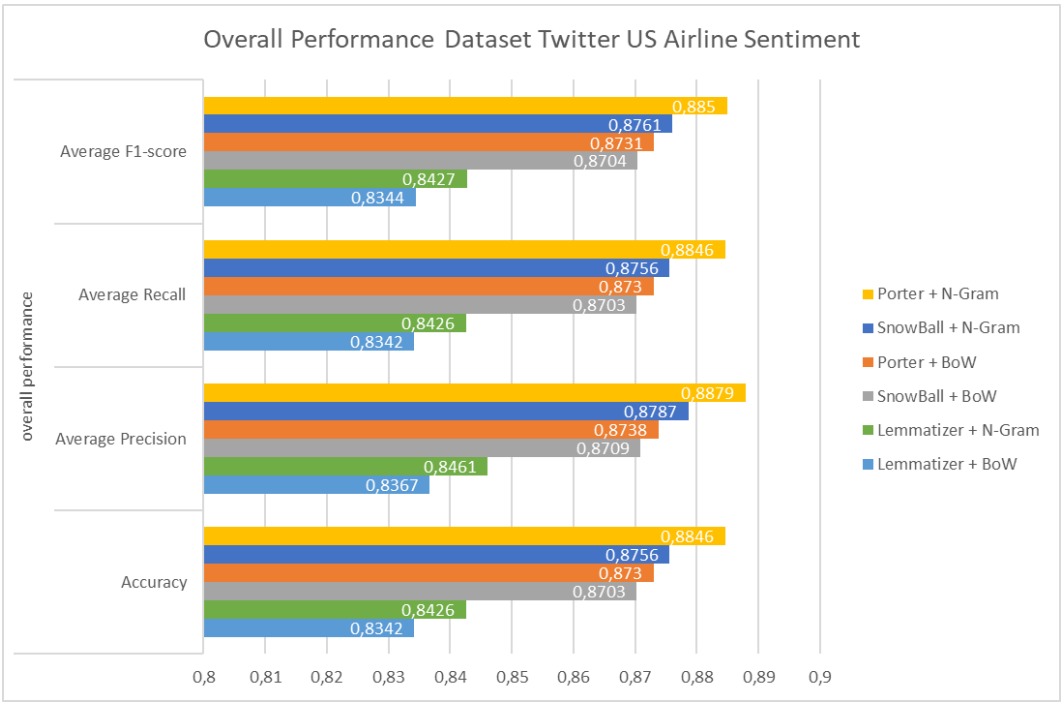
Gambar 4. 46 Grafik Evaluasi Model LiniearSVC Dataset 1

Tabel 4. 55 dan Gambar 4.46 menunjukkan hasil evaluasi model baik *accuracy*, *precision*, *recall* dan *F1-Score* dari setiap skenario gabungan *stemmer* dan *feature extraction* pada dataset 1 yakni IMDB dataset. Gabungan antara

SnowBall dan BoW memberikan nilai *accuracy, precision, recall* dan *F1-Score* terrendah. Sedangkan gabungan antara *SnowBall* dan BoW memberikan nilai *accuracy, precision, recall* dan *F1-Score* memberikan hasil terbaik dengan nilai akurasi sebesar 96,08 %. *Lemmatizer* mengubah kata ke dalam bentuk dasarnya (lemma) sehingga membantu model mengidentifikasi kata-kata dengan makna sempurna dan N-Gram membantu model mamahami konteks dan makna kalimat dengan nurutan kata dalam teks. Sehingga gabungan antara keduanya memberikan hasil terbaik pada implementasi model.

Tabel 4. 56 Hasil Evaluasi Model LiniearSVC Dataset 2

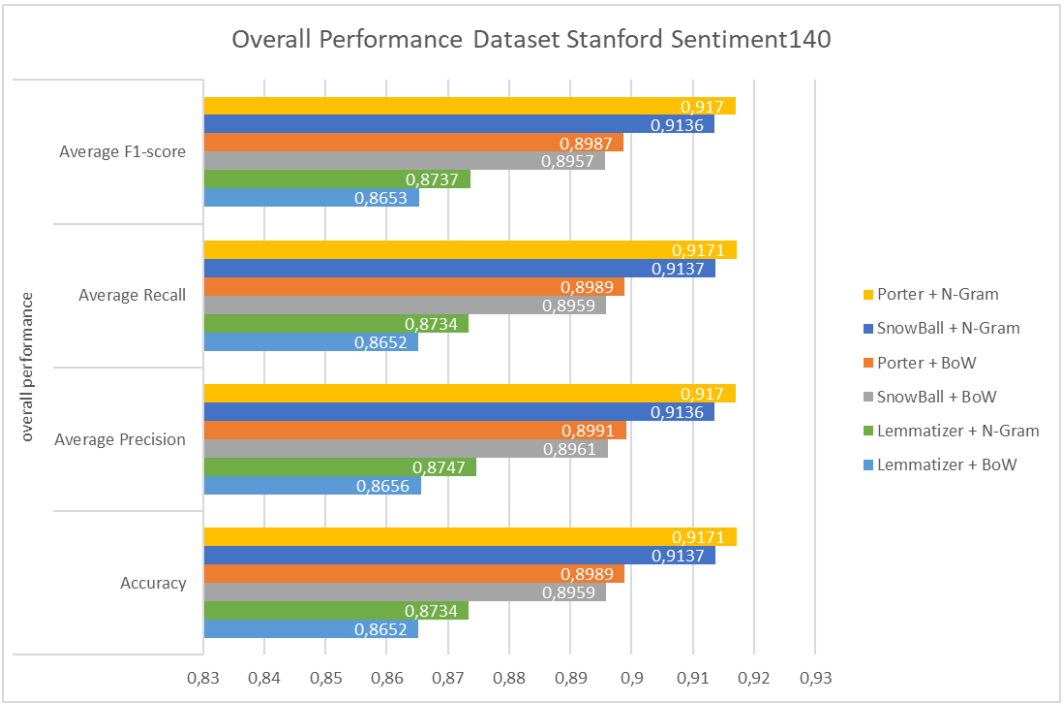
Twitter US Airline Sentimen	Overall Performance			
	Accuracy	Average Precision	Average Recall	Average F1-score
Porter + N-Gram	0,8846	0,8879	0,8846	0,8850
Porter + BoW	0,8730	0,8738	0,8730	0,8731
SnowBall + N-Gram	0,8756	0,8787	0,8756	0,8761
SnowBall + BoW	0,8703	0,8709	0,8703	0,8704
Lemmatizer + N-Gram	0,8426	0,8461	0,8426	0,8427
Lemmatizer + BoW	0,8342	0,8367	0,8342	0,8344



Gambar 4. 47 Grafik Evaluasi Model LiniearSVC Dataset 2

Tabel 4. 57 Hasil Evaluasi Model LiniearSVC Dataset 3

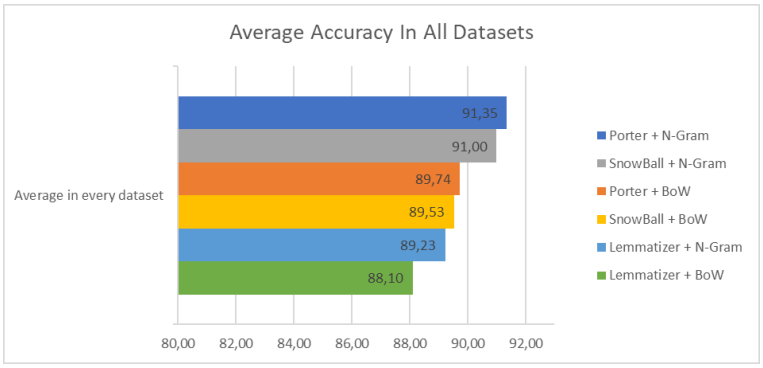
Stanford sentiment140	Overall Performance			
	Accuracy	Average Precision	Average Recall	Average F1-score
Porter + N-Gram	0,9171	0,9170	0,9171	0,9170
Porter + BoW	0,8989	0,8991	0,8989	0,8987
SnowBall + N-Gram	0,9137	0,9136	0,9137	0,9136
SnowBall + BoW	0,8959	0,8961	0,8959	0,8957
Lemmatizer + N-Gram	0,8734	0,8747	0,8734	0,8737
Lemmatizer + BoW	0,8652	0,8656	0,8652	0,8653



Gambar 4. 48 Grafik Evaluasi Model LiniearSVC Dataset 3

Sedangkan pada Tabel 4. 56 serta Gambar 4.47 (Hasil evaluasi model pada dataset Twitter US Airline Sentimen) dan Tabel 4. 57 serta Gambar 4.48 (Hasil evaluasi model pada dataset Stanford sentiment140) menunjukkan hasil evaluasi model baik *accuracy*, *precision*, *recall* dan *F1-Score* terbaik ada pada gabungan antara *Porter* dan N-Gram. Sedangkan gabungan antara *Lemmatizer* dan BoW memberikan nilai *accuracy*, *precision*, *recall* dan *F1-Score* memberikan hasil paling rendah. *Poter* termasuk *stemmer* yang cukup efektif populer dalam bahasa Inggris. Serta cukup efektif dalam menghapus awalan dan akhiran kata dengan tetap menjaga informasi penting. Dan N-Gram menangkap urutan kata dalam teks, sehingga membantu model memahami makna dan konteks pada kalimat.

Secara keseluruhan dapat dipahamin berdasarkan hasil evaluasi model yang dipaparkan, gabungan antara *stemmer* dan N-Gram mendapatkan nilai yang lebih baik dibandingkan dengan gabungan antara *stemmer* dan BoW. Karena N-Gram lebih efektif dibandingkan BoW dalam mengekstrak fitur. Rata-rata akurasi dari gabungan *stemmer* dan *feature extractio*n terbaik berdasarkan hasil evaluasi model di setiap dataset ada pada *Porter* dan N-Gram, seperti yang dipaparkan pada Gambar 4. 49.



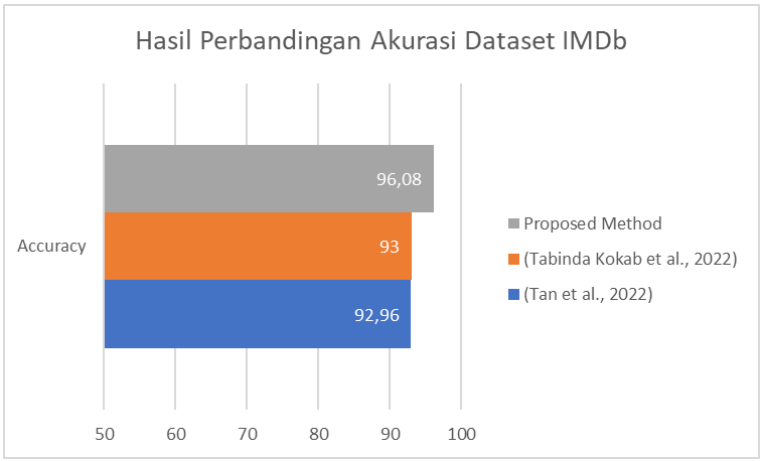
Gambar 4. 49 Rata-rata Akurasi Gabungan *Stemmer* dan *Feature Extraction*

Penelitian ini akan membandingkan hasil penelitian dengan penelitian-penelitian sebelumnya yang menggunakan dataset yang sama yaitu IMDb, Twitter US Airline Sentiment, dan Sentiment140 dalam kasus analisis sentimen. Perbandingan dilakukan untuk menunjukkan kontribusi dan signifikansi penelitian ini terhadap pengetahuan yang ada. Karna dipandang penting untuk membandingkan hasil penelitian ini dengan penelitian sebelumnya yang menggunakan dataset yang sama untuk memastikan bahwa hasil yang diperoleh valid dan dapat digeneralisasikan. Berikut adalah perbandingan hasil penelitian

dengan penelitian-penelitian sebelumnya berdasarkan dataset yang digunakan ditunjukkan pada Tabel 4. 58 – 4. 60.

Tabel 4. 58 Hasil Perbandingan Akurasi Dataset IMDb

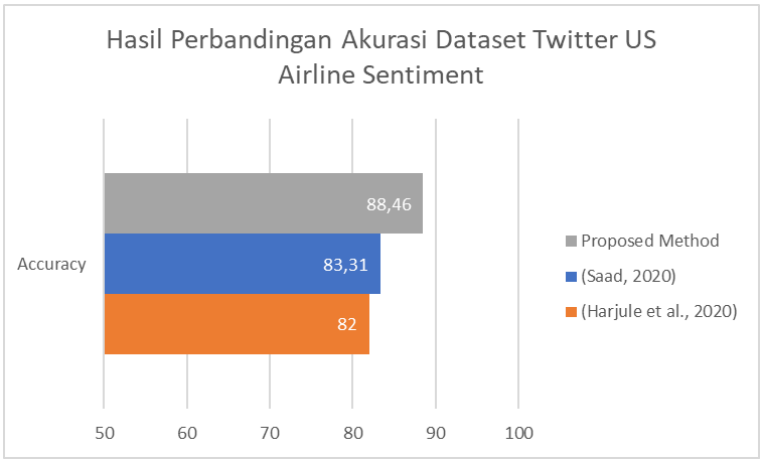
Literature	Dataset	Model	Accuracy %
(Tan et al., 2022)	IMDb	RoBERTa-LSTM	92.96
(Tabinda Kokab et al., 2022)	IMDb	CBRNN	93
Proposed Method	IMDb	SVM (LinearSVC)	96,08



Gambar 4. 50 Hasil Perbandingan Akurasi Dataset IMDb

Tabel 4. 59 Hasil Perbandingan Akurasi Dataset Twitter US Airline Sentiment

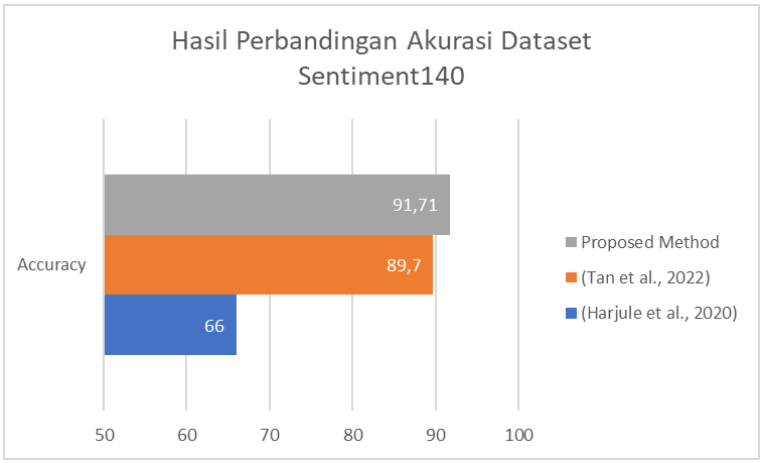
Literature	Dataset	Model	Accuracy %
(Saad, 2020)	Twitter US Airline Sentiment	SVM	83,31
(Harjule et al., 2020)	Twitter US Airline Sentiment	LSTM	82
Proposed Method	Twitter US Airline Sentiment	SVM (LinearSVC)	88,46



Gambar 4. 51 Hasil Perbandingan Akurasi Dataset Twitter US Airline Sentiment

Tabel 4. 60 Hasil Perbandingan Akurasi Dataset Sentiment140

Literature	Dataset	Model	Accuracy %
(Harjule et al., 2020)	Sentiment140	LSTM	66
(Tan et al., 2022)	Sentiment140	RoBERTa-LSTM	89,7
Proposed Method	Sentiment140	SVM (LinearSVC)	91,71



Gambar 4. 52 Hasil Perbandingan Akurasi Dataset Sentiment140

Dari perbandingan yang dipaparkan dapat disimpulkan bahwasanya, penelitian sebelumnya telah menunjukkan bahwa akurasi model klasifikasi teks

khususnya sentimen analisis dapat berfariasi. Hal ini dapat disebabkan oleh beberapa faktor, seperti karakteristik dataset, metode yang digunakan dan parameter pada model. Penelitian ini juga menghasilkan akurasi yang lebih baik dibandingkan dengan penelitian sebelumnya pada setiap dataset yang digunakan. Hal ini menunjukkan bahwa metode yang diusulkan dalam penelitian ini dipandang efektif dalam mengklasifikasi teks sentiment. Dibuktikan dengan akurasi pada dataset IMDb dengan paduan *Lemmatizer* dan N-Gram, Twitter US Airline Sentiment, dan Sentiment140 dengan paduan *Porter* dan N-Gram. Masing-masing mencatat akurasi 96,08 %, 88,46 % dan 91,71 %.

BAB V

PENUTUP

5.1. Kesimpulan

Setelah melakukan eksperiment terkait pengaruh *stemmer* dan *feature extraction* terhadap algoritma SVM pada analisis sentimen berbasis leksikon. Dengan melakukan beberapa skenario antara gabungan *stemmer* dan *feature extraction* pada 3 dataset yang berbeda, dapat diambil beberapa kesimpulan seperti berikut:

4. *Stemmer* dan *feature extraction* berpengaruh terhadap nilai *accuracy, precision, recall* dan *F1-Score* pada algoritma *Support Vector Machine* (SVM) dalam analisis sentimen berbasis leksikon, dibuktikan berdasarkan implementasi gabungan keduanya pada tiga dataset yang berbeda. Nilai *accuracy, precision, recall* dan *F1-Score* rata-rata diatas 80 %.
5. *Porter Stemmer* dan N-Gram merupakan gabungan *stemmer* dan *feature extraction* yang paling efektif meningkatkan kinerja SVM, dengan rata-rata akurasi 91, 35 % pada tiga dataset yang digunakan.
6. Metode yang diusulkan dalam penelitian ini dipandang efektif dalam mengklasifikasi teks sentiment. Dibuktikan dengan akurasi pada dataset IMDb dengan paduan *Lemmatizer* dan N-Gram, Twitter US Airline Sentiment, dan Sentiment140 dengan paduan *Porter* dan N-Gram. Masing-masing mencatat akurasi 96,08 %, 88,46 % dan 91,71 %.

5.2. Saran

Berdasarkan kesimpulan yang dipaparkan, penelitian ini masih memiliki kekurangan dan kendala, maka akan lebih baik pada penelitian selanjutnya dapat memperbaiki kekurangan yang ada. Berikut beberapa saran untuk penelitian selanjutnya:

1. Mengoptimalkan *cleaning*, dengan menambahkan fungsi untuk memperbaiki kesalahan pada *cleaning*, karena akan berpengaruh pada hasil dari *stemming*.
2. Dibutuhkan kajian ulang atau perbandingan atara *preprocessing* yang lengkap dan tidak lengkap seperti yang dilakukan pada penelitian oleh (Rustam et al., 2019).
3. Melakukan improvement pada bagian *feature extraction* dengan menggunakan *feature extraction* terbaru seperti, Glove, Word2Vec, Doc2Vec dan lain sebagainya.
4. Melakukan pengujian terhadap data dengan dan tanpa handling imbalance.
5. Menggunakan model *machine learning* yang berbeda untuk memperkuat analisa tentang perpaduan *stemmer* dan *feature extraction*.

DAFTAR PUSTAKA

PUSTAKA MAJALAH, JURNAL ILMIAH ATAU PROSIDING

Ahmed, K., Nadeem, M. I., Li, D., Zheng, Z., Ghadi, Y. Y., Assam, M., & Mohamed, H. G. (2022). Exploiting Stacked Autoencoders for Improved Sentiment Analysis. *Applied Sciences (Switzerland)*, *12*(23). https://doi.org/10.3390/app122312380

Ahuja, R., Chug, A., Kohli, S., Gupta, S., & Ahuja, P. (2019). The impact of features extraction on the sentiment analysis. *Procedia Computer Science*, *152*, 341–348. https://doi.org/10.1016/j.procs.2019.05.008

AlBadani, B., Shi, R., & Dong, J. (2022). A Novel Machine Learning Approach for Sentiment Analysis on Twitter Incorporating the Universal Language Model Fine-Tuning and SVM. *Applied System Innovation*, *5*(1). https://doi.org/10.3390/asi5010013

Ayu, M. A., Wijaya, S. S., & Mantoro, T. (2019). An automatic lexicon generation for Indonesian news sentiment analysis: A case on governor elections in Indonesia. *Indonesian Journal of Electrical Engineering and Computer Science*, *16*(3), 1555–1561. https://doi.org/10.11591/ijeecs.v16.i3.pp1555-1561

Babu, N. V., & Kanaga, E. G. M. (2022). Sentiment Analysis in Social Media Data for Depression Detection Using Artificial Intelligence: A Review. *SN Computer Science*, *3*(1), 1–20. https://doi.org/10.1007/s42979-021-00958-1

Barve, Y., Saini, J. R., Pal, K., & Kotecha, K. (2022). A Novel Evolving Sentimental Bag-of-Words Approach for Feature Extraction to Detect Misinformation. *International Journal of Advanced Computer Science and Applications*, *13*(4), 266–275. https://doi.org/10.14569/IJACSA.2022.0130431

Chen, B., Huang, Q., Chen, Y., Cheng, L., & Chen, R. (2019). Deep Neural Networks for Multi-class Sentiment Classification. *Proceedings - 20th International Conference on High Performance Computing and Communications, 16th International Conference on Smart City and 4th International Conference on Data Science and Systems, HPCC/SmartCity/DSS 2018*, 854–859. https://doi.org/10.1109/HPCC/SmartCity/DSS.2018.00142

Dangi, D., Bhagat, A., & Dixit, D. K. (2022). Sentiment analysis of social media data based on chaotic coyote optimization algorithm based time weight-AdaBoost support vector machine approach. In *Concurrency and*

Computation: Practice and Experience (Vol. 34, Issue 3). https://doi.org/10.1002/cpe.6581

Daniel Jurafsky & James H. Martin. (2006). *Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition*. papers3://publication/uuid/531A3835-7600-4448-853F-34C2CDA40E8D

David L. Olson, & Delen, D. (2008). Advanced Data Mining Techniques. In *Angewandte Chemie International Edition, 6(11), 951–952.* (Issue Mi).

Fellbaum, C. (1998). WordNet: An Electronic Lexical Database. *MA: MIT Press*. https://doi.org/10.2307/417141

Ferdiana, R., Fajar, W., Purwanti, D. D., Ayu, A. S. T., & Jatmiko, F. (2019). Twitter sentiment analysis in under-resourced languages using byte-level recurrent neural model. *International Journal of Advanced Computer Science and Applications*, *10*(8), 108–112. https://doi.org/10.14569/ijacsa.2019.0100815

Fikri, M., & Sarno, R. (2019). A comparative study of sentiment analysis using SVM and Senti Word Net. *Indonesian Journal of Electrical Engineering and Computer Science*, *13*(3), 902–909. https://doi.org/10.11591/ijeecs.v13.i3.pp902-909

Fitriyyah, S. N. J., Safriadi, N., & Pratama, E. E. (2019). Analisis Sentimen Calon Presiden Indonesia 2019 dari Media Sosial Twitter Menggunakan Metode Naive Bayes. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, *5*(3), 279. https://doi.org/10.26418/jp.v5i3.34368

Gowri, S., Surendran, R., Divya Bharathi, M., & Jabez, J. (2022). Improved Sentimental Analysis to the Movie Reviews using Naive Bayes Classifier. *Proceedings of the International Conference on Electronics and Renewable Systems, ICEARS 2022, May*, 1831–1836. https://doi.org/10.1109/ICEARS53579.2022.9752408

Han, K. X., Chien, W., Chiu, C. C., & Cheng, Y. T. (2020). Application of support vector machine (SVM) in the sentiment analysis of twitter dataset. *Applied Sciences (Switzerland)*, *10*(3). https://doi.org/10.3390/app10031125

Handayani, Y., Hakim, A. R., & Muljono. (2020). Sentiment analysis of Bank BNI user comments using the support vector machine method. *Proceedings - 2020 International Seminar on Application for Technology of Information and Communication: IT Challenges for Sustainability, Scalability, and Security in the Age of Digital Disruption, ISemantic 2020*, 202–207. https://doi.org/10.1109/iSemantic50169.2020.9234230

Harjule, P., Gurjar, A., Seth, H., & Thakur, P. (2020). Text Classification on Twitter Data. *Proceedings of 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things, ICETCE 2020*, *February*, 160–164. https://doi.org/10.1109/ICETCE48199.2020.9091774

Harris, Z. S. (1954). Distributional Structure. WORD, *10*(2–3), 146–162. https://doi.org/10.1080/00437956.1954.11659520

Jones, K. S. (2004). A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. *Journal of Documentation*.

Kalangi, R. R., Maloji, S., Tejasri, N., Chand, P. P., & Ch, V. P. (2021). Sentiment Analysis using Machine Learning. *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 116–121. https://doi.org/10.1109/ICAC3N53548.2021.9725499

Kumar, V., & Subba, B. (2020). A tfidfvectorizer and SVM based sentiment analysis framework for text data corpus. *26th National Conference on Communications, NCC 2020*, 1–6. https://doi.org/10.1109/NCC48643.2020.9056085

Liu, B. (2012). *Sentiment Analysis and Mining of Opinions* (Issue April). Morgan & Claypool Publishers. https://doi.org/10.1007/978-3-319-60435-0_20

Manning, C. D., Schütze, H., & Weikurn, G. (2002). Foundations of Statistical Natural Language Processing. *SIGMOD Record*, *31*(3), 37–38. https://doi.org/10.1145/601858.601867

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, *January 2013*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations ofwords and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 1–9.

Muhammad, A., Abdullah, S., & Sani, N. S. (2021). Optimization of sentiment analysis using teaching-learning based algorithm. *Computers, Materials and Continua*, *69*(2), 1783–1799. https://doi.org/10.32604/cmc.2021.018593

Nurcahyawati, V., & Mustaffa, Z. (2023). Improving sentiment reviews classification performance using support vector machine-fuzzy matching algorithm. *Bulletin of Electrical Engineering and Informatics*, *12*(3), 1817–1824. https://doi.org/10.11591/eei.v12i3.4830

Obiedat, R., Qaddoura, R., Al-Zoubi, A. M., Al-Qaisi, L., Harfoushi, O., Alrefai,

M., & Faris, H. (2022). Sentiment Analysis of Customers' Reviews Using a Hybrid Evolutionary SVM-Based Approach in an Imbalanced Data Distribution. *IEEE Access*, *10*, 22260–22273. https://doi.org/10.1109/ACCESS.2022.3149482

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, *January*, 1532–1543. https://doi.org/10.3115/v1/d14-1162

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, *14*(3), 130–137.

Porter, M. F. (1996). *Snowball: A language for stemming algorithms*. http://snowball.tartarus.org/texts/introduction.html

Prastyo, P. H., Sumi, A. S., Dian, A. W., & Permanasari, A. E. (2020). Tweets Responding to the Indonesian Government's Handling of COVID-19: Sentiment Analysis Using SVM with Normalized Poly Kernel. *Journal of Information Systems Engineering and Business Intelligence*, *6*(2), 112. https://doi.org/10.20473/jisebi.6.2.112-122

Qaisar, S. M. (2020). Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory. *2020 2nd International Conference on Computer and Information Sciences, ICCIS 2020*, 12–15. https://doi.org/10.1109/ICCIS49240.2020.9257657

Qi, Y., & Shabrina, Z. (2023). Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach. In *Social Network Analysis and Mining* (Vol. 13, Issue 1). https://doi.org/10.1007/s13278-023-01030-x

Rahat, A. M., Kahir, A., Kaisar, A., & Masum, M. (2019). *Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset*.

Rajesh, P., & Suseendran, G. (2020). Prediction of N-Gram Language Models Using Sentiment Analysis on E-Learning Reviews. *2020 International Conference on Intelligent Engineering and Management (ICIEM)*, 510–514. https://doi.org/10.1109/ICIEM48762.2020.9160260

Rani, S., Singh Gill, N., & Gulia, P. (2021). 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. *Indonesian Journal of Electrical Engineering and Computer Science*, *22*(2), 1041. https://doi.org/10.11591/ijeecs.v22.i2.pp1041-1051

Ressan, M. B., & Hassan, R. F. (2022). Naïve-Bayes family for sentiment analysis during COVID-19 pandemic and classification tweets. *Indonesian Journal of*

Electrical Engineering and Computer Science, *28*(1), 375–383. https://doi.org/10.11591/ijeecs.v28.i1.pp375-383

Resyanto, F., Sibaroni, Y., & Romadhony, A. (2019). Choosing The Most Optimum Text Preprocessing Method for Sentiment Analysis: Case:iPhone Tweets. *Proceedings of 2019 4th International Conference on Informatics and Computing, ICIC 2019*, 2–6. https://doi.org/10.1109/ICIC47613.2019.8985943

Rustam, F., Ashraf, I., Mehmood, A., Ullah, S., & Choi, G. S. (2019). Tweets classification on the base of sentiments for US airline companies. *Entropy*, *21*(11), 1–22. https://doi.org/10.3390/e21111078

Rustam, F., Khalid, M., Aslam, W., Rupapara, V., Mehmood, A., & Choi, G. S. (2021). A performance comparison of supervised machine learning models for Covid-19 tweets sentiment analysis. *PLoS ONE*, *16*(2), 1–23. https://doi.org/10.1371/journal.pone.0245909

Saad, A. I. (2020). Opinion Mining on US Airline Twitter Data Using Machine Learning Techniques. *16th International Computer Engineering Conference, ICENCO 2020*, 59–63. https://doi.org/10.1109/ICENCO49778.2020.9357390

SALTON, G., & BUCKLEY, C. (1988). TERM-WEIGHTING APPROACHES IN AUTOMATIC TEXT RETRIVAL. *Information Processing & Management*, *24*(3), 513–523. https://doi.org/10.1163/187631286X00251

Semary, N. A., Ahmed, W., Amin, K., Pławiak, P., & Hammad, M. (2024). Enhancing machine learning-based sentiment analysis through feature extraction techniques. *PLoS ONE*, *19*(2 February). https://doi.org/10.1371/journal.pone.0294968

Sethi, M., Pandey, S., Trar, P., & Soni, P. (2020). Sentiment Identification in COVID-19 Specific Tweets. *Proceedings of the International Conference on Electronics and Sustainable Communication Systems, ICESC 2020*, *Icesc*, 509–516. https://doi.org/10.1109/ICESC48915.2020.9155674

Shaddeli, A., Soleimanian Gharehchopogh, F., Masdari, M., & Solouk, V. (2022). An Improved African Vulture Optimization Algorithm for Feature Selection Problems and Its Application of Sentiment Analysis on Movie Reviews. In *Big Data and Cognitive Computing* (Vol. 6, Issue 4). https://doi.org/10.3390/bdcc6040104

Shamrat, F. M. J. M., Chakraborty, S., Imran, M. M., Muna, J. N., Billah, M. M., Das, P., & Rahman, M. O. (2021). Sentiment analysis on twitter tweets about COVID-19 vaccines using NLP and supervised KNN classification algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*, *23*(1), 463–470. https://doi.org/10.11591/ijeecs.v23.i1.pp463-470

Shofiya, C., & Abidi, S. (2021). Sentiment analysis on covid-19-related social distancing in Canada using twitter data. *International Journal of Environmental Research and Public Health*, *18*(11). https://doi.org/10.3390/ijerph18115993

Shuai, Z., Xiaolin, D., Jing, Y., Yanni, H., Meng, C., Yuxin, W., & Wei, Z. (2022). Comparison of different feature extraction methods for applicable automated ICD coding. *BMC Medical Informatics and Decision Making*, *22*(1), 1–15. https://doi.org/10.1186/s12911-022-01753-5

Suryati, E., Styawati, & Ari Aldino, A. (2023). Analisis Sentimen Transportasi Online Menggunakan Ekstraksi Fitur Model Word2vec Text Embedding Dan Algoritma Support Vector Machine (SVM). *Jurnal Teknologi Dan Sistem Informasi*, *4*(1), 96–106.

Tabinda Kokab, S., Asghar, S., & Naz, S. (2022). Transformer-based deep learning models for the sentiment analysis of social media data. *Array*, *14*(April), 100157. https://doi.org/10.1016/j.array.2022.100157

Tamara, K., & Milićević, N. (2018). Comparing Sentiment Analysis and Document Representation Methods of Amazon Reviews. *SISY 2018 - IEEE 16th International Symposium on Intelligent Systems and Informatics, Proceedings*, 283–288. https://doi.org/10.1109/SISY.2018.8524814

Tan, K. L., Lee, C. P., Lim, K. M., & Anbananthen, K. S. M. (2022). Sentiment Analysis With Ensemble Hybrid Deep Learning Model. *IEEE Access*, *10*(July), 103694–103704. https://doi.org/10.1109/ACCESS.2022.3210182

Verma, P., Dumka, A., Bhardwaj, A., & Ashok, A. (2022). Product Review-Based Customer Sentiment Analysis Using an Ensemble of mRMR and Forest Optimization Algorithm (FOA). *International Journal of Applied Metaheuristic Computing*, *13*(1), 1–21. https://doi.org/10.4018/ijamc.2022010107

Zipf, G. K. (1999). *The Psycho-Biology of Language: An Introduction to Dynamic Philology*. Psychology Press. https://books.google.co.id/books?id=w1Z4Aq-5sWMC&lpg=PP1&hl=id&pg=PP1#v=onepage&q&f=false

Zou, H., Tang, X., Xie, B., & Liu, B. (2016). Sentiment classification using machine learning techniques with syntax features. *Proceedings - 2015 International Conference on Computational Science and Computational Intelligence, CSCI 2015*, 175–176. https://doi.org/10.1109/CSCI.2015.44

LAMPIRAN