ANALISIS SENTIMEN TERHADAP SIMPANAN TAPERA DI INSTAGRAM MENGGUNAKAN METODE WORD2VEC DENGAN ALGORITMA NAIVES BAYES DAN SMOTE

diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana pada program studi Informatika.

SKRIPSI

Oleh

Anggik Muhammad Sahril
182410103021

KEMENTERIAN PENDIDIKAN, TINGGI, SAINS DAN TEKNOLOGI
UNIVERSITAS JEMBER
FAKULTAS ILMU KOMPUTER
INFORMATIKA
JEMBER
2025

PERSEMBAHAN

Alhamdulillah, puji syukur atas kehadirat Allah SWT yang selalu memberikan rahmat dan hidayahnya, sehingga skripsi ini dapat diselesaikan. Dengan penuh kerendahan hati, skripsi ini saya persembahkan untuk:

1. Ibunda tercinta Dewi dan Kakak terdekat Melina.
2. Keluarga besar.
3. Guru-guru saya sejak dibangku sekolah dasar sampai dosen perguruan tinggi.
4. Almamater Program Studi Informatika Fakultas Ilmu Komputer Universitas Jember.

MOTTO

“Keimanan yang akan membuatmu selalu bertahan
dengan apapun proses yang dilewati”
-Anggik M.S-

“Sesungguhnya kami milik Allah dan kepada-Nyalah kani kenbali”
-QS. Al-Baqarah 156-

PERNYATAAN ORISINALITAS

PERNYATAAN ORISINALITAS

Saya yang bertanda tangan di bawah ini :

Nama : Anggik Muhammad Sahril

NIM : 182410103021

Menyatakan dengan sesungguhnya bahwa skripsi yang berjudul: Analisis Sentimen terhadap Simpanan Tapera di Instagram Menggunakan Metode Word2Vec dengan Algoritma Naives Bayes dan SMOTE.

adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, dan belum pernah diajukan pada institusi manapun, serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, 24 Juni 2025

Yang menyatakan,

Anggik Muhammad Sahril

NIM 182410103021

iv

HALAMAN PERSETUJUAN

HALAMAN PERSETUJUAN

Skripsi berjudul Analisis Sentimen terhadap Simpanan Tapera di Instagram Menggunakan Metode Word2Vec dengan Algoritma Naives Bayes dan SMOTE. telah diuji dan disetujui oleh Fakultas Ilmu Komputer Universitas Jember pada:

Hari : Senin
Tanggal : 23 Juni 2025
Tempat : Fakultas Ilmu Komputer Universitas Jember

Pembimbing — Tanda Tangan

1. Pembimbing Utama
   Nama : Muhamad Arief Hidayat S.Kom,.M.Kom. (.....)
   NIP : 198101232010121003
2. Pembimbing Anggota
   Nama : Qurota A'Yuni Ar Rahmat, S.Pd.M.Sc. (.....)
   NIP : 760018029

Penguji

1. Penguji Utama
   Nama : Priza Pandunata S.Kom., M.Sc. (.....)
   NIP : 198301312015041001
2. Penguji Anggota 1
   Nama : Muhammad 'Ariful Furqon, S.Pd., M.Kom.(.....)
   NIP : 199407262020121005

v

ABSTRAK

Tabungan Perumahan Rakyat (TAPERA) menuai berbagai respons masyarakat di Instagram. Penelitian ini bertujuan menganalisis sentimen pengguna Instagram terhadap TAPERA menggunakan metode Word2Vec dengan algoritma Naive Bayes dan SMOTE, serta mengevaluasi akurasi yang dihasilkan algoritma SMOTE. Penelitian kuantitatif ini mengumpulkan data komentar Instagram tentang TAPERA yang diproses menggunakan Word2Vec untuk representasi kata, dikombinasikan dengan Naive Bayes untuk klasifikasi sentimen, dan SMOTE untuk mengatasi ketidakseimbangan dataset. Hasil menunjukkan model memiliki akurasi 55% dengan performa kurang optimal akibat ketidakseimbangan dataset, dimana data negatif (157 sampel) jauh lebih banyak dari data positif (43 sampel). Model menunjukkan bias terhadap prediksi negatif dengan precision 83%, tetapi lemah memprediksi positif dengan precision 26%. Penelitian menyimpulkan bahwa kombinasi Word2Vec, Naive Bayes, dan SMOTE dapat digunakan untuk analisis sentimen TAPERA, namun performa model memerlukan perbaikan melalui penyeimbangan dataset, penyetelan parameter, dan pemilihan metrik evaluasi yang tepat.

Kata Kunci**:** analisis sentimen, TAPERA, Word2Vec, Naive Bayes, SMOTE, Instagram, ketidakseimbangan dataset

ABSTRACT

Tabungan Perumahan Rakyat (TAPERA) program has generated various public responses on Instagram. This research aims to analyze Instagram users' sentiment toward TAPERA using the Word2Vec method with Naive Bayes and SMOTE algorithms, as well as evaluate the accuracy produced by the SMOTE algorithm. This quantitative research collected Instagram comment data about TAPERA, which was processed using Word2Vec for word representation, combined with Naive Bayes for sentiment classification, and SMOTE to address dataset imbalance. The results show that the model has 55% accuracy with suboptimal performance due to dataset imbalance, where negative data (157 samples) far exceeds positive data (43 samples). The model shows bias toward negative predictions with 83% precision, but performs poorly in predicting positive sentiment with only 26% precision. The research concludes that the combination of Word2Vec, Naive Bayes, and SMOTE can be used for TAPERA sentiment analysis, however, model performance requires improvement through better dataset balancing, parameter tuning, and appropriate evaluation metric selection.

*Keywords***:** *sentiment analysis, TAPERA, Word2Vec, Naive Bayes, SMOTE, Instagram, dataset imbalance*

RINGKASAN

Analisis Sentimen terhadap Simpanan Tapera di Instagram Menggunakan Metode Word2Vec dengan Algoritma Naives Bayes dan SMOTE ; Anggik Muhammad Sahril, 182410103021; 31 Halaman, Program Studi Informatika Fakultas Ilmu Komputer Universitas Jember

Defisit perumahan di Indonesia, khususnya di wilayah perkotaan, masih menjadi permasalahan utama yang dihadapi masyarakat berpenghasilan rendah. Pemerintah merespons hal ini dengan kebijakan Tabungan Perumahan Rakyat (TAPERA) yang mewajibkan iuran sebesar 3% dari penghasilan pekerja, dengan rincian 0,5% ditanggung oleh pemberi kerja dan 2,5% oleh pekerja. Meskipun bertujuan mengatasi kekurangan perumahan, kebijakan ini menuai pro dan kontra.

Seiring perkembangan teknologi, media sosial seperti Instagram menjadi wadah masyarakat untuk menyuarakan pendapat mereka terhadap kebijakan pemerintah, termasuk TAPERA. Untuk menganalisis sentimen dari data teks yang dihasilkan, digunakan pendekatan canggih seperti word embedding, salah satunya Word2Vec. Teknik ini mengubah kata-kata menjadi vektor numerik dalam ruang multidimensi, sehingga memungkinkan pemahaman hubungan semantik antar kata. Representasi ini sangat berguna dalam berbagai tugas pemrosesan bahasa alami seperti klasifikasi teks dan analisis sentimen.

Dalam penelitian ini, algoritma Naive Bayes dipilih sebagai metode utama untuk klasifikasi sentimen karena kemampuannya yang cukup efektif dalam mengelompokkan data ke dalam kategori positif, negatif, dan netral. Namun, karena data sering kali tidak seimbang, algoritma SMOTE juga digunakan untuk mengatasi ketidakseimbangan antara jumlah data positif dan negatif. Penggunaan metode ini merujuk pada berbagai penelitian terdahulu yang telah membuktikan efektivitas kombinasi Word2Vec dan algoritma klasifikasi seperti Naive Bayes maupun Support Vector Machine dalam menganalisis opini publik melalui media sosial dan ulasan aplikasi digital.

PRAKATA

Puji syukur kehadirat Allah SWT. Atas segala rahmat dan hidayahnya, sehingga penulis dapat menyelesaikan skripsi dengan judul "Analisis Sentimen terhadap Simpanan Tapera di Instagram Menggunakan Metode Word2Vec dengan Algoritma Naives Bayes dan SMOTE". Skripsi ini disusun untuk memenuhi salahsatu syarat menyelesaikan Pendidikan Strata Satu (S1) pada Program Studi Informatika Fakultas Ilmu Komputer Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan ucapan Terima Kasih Kepada :

1. Allah SWT yang senentiasa memberikan rahmat dan hidayahnya untuk mempermudah dan melancarkan dalam mengerjakan skripsi;
2. Dosen Pembimbing Utama, Bapak Muhamad Arief Hidayat S.Kom,.M.Kom. yang telah memberikan arahan, nasehat, ilmu, saran dan koreksi dengan penuh kesabaran;
3. Dosen Pembimbing Pendamping, Ibu Qurrota A"yuni Ar Ruhimat S.Pd., M.Sc. yang telah memberikan arahan, nasehat, ilmu, saran dan koreksi dengan penuh kesabaran;
4. Bapak Priza Pandunata S.Kom., M.Sc. Selaku Dosen Penguji Utama dan Bapak Muhammad `Ariful Furqon, S.Pd., M.Kom. selaku Dosen Penguji Pendamping yang telah berkenan untuk menguji skripsi ini dan memberikan masukan dan saran untuk menyempurnakan skripsi ini;
5. Bapak Drs. Antonius Cahya Prihandoko, M.App.Sc, Ph.D selaku dekan Fakultas Ilmu Komputer Universitas Jember;
6. Seluruh Bapak dan Ibu dosen beserta staff karyawan di Program Studi Informatika Fakultas Ilmu Komputer Universitas Jember;
7. Kedua Orang tua dan seluruh keluarga saya yang telah memberikan do'a dan dukungan setiap harinya;

8. Teman-teman terdekat Renanta dan Fendik;
9. Keluarga besar UKM BALWANA;
10. Keluarga besar Program Studi Informatika Fakultas Ilmu Komputer Universitas Jember Angkatan 2018;
11. Keluarga besar Fakultas Ilmu Komputer Universitas Jember;
12. Sahabat-sahabat saya yang telah banyak memberikan semangat, membantu dan menemani dalam pengerjaan skripsi ini;
13. Semua pihak yang telah membantu mensukseskan skripsi ini yang tidak dapat saya sebutkan satu-persatu;

Dengan harapan penelitia ini akan selalu berkembang. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, oleh karena ini penulis mengharapkan adanya masukan yang bersifat membangun dari semua pihak. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak.

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR LAMPIRAN

DAFTAR ISTILAH DAN SINGKATAN

Singkatan/Istikal	Arti dan keterangan
TAPERA	Tabungan Perumahan Rakyat
SMOTE	Sythetic Minority Oversampling Technique
API	Application Programming Interface
SVM	Support Vector Model
NLP	Natural Language Processing

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Di Indonesia, defisit perumahan masih menjadi permasalahan yang dihadapi oleh masyarakat, terutama di perkotaan. Menurut data Kementerian Pekerjaan Umum dan Perumahan Rakyat, defisit perumahan mencapai jutaan unit, dengan mayoritas terjadi di segmen masyarakat berpenghasilan rendah. “Defisit ini tidak hanya mempengaruhi akses masyarakat terhadap perumahan yang layak, tetapi juga berdampak pada stabilitas sosial dan ekonomi di tingkat local maupun nasional.” (Herry TZ Dirjen Pekerjaan Umum, 2024). Maka dari itu, munculah Kebijakan Tabungan Perumahan Rakyat (TAPERA) merupakan inisiatif pemerintah Indonesia untuk mengatasi masalah defisit perumahan ini. Adapun beberapa hal yang diperhatikan terkait dengan TAPERA yaitu. Pertama, Pasal 15 PP TAPERA mengatur besaran simpanan peserta ditetapkan sebesar 3% dari gaji atau upah pekerja. Besaran itu dibayarkan 0,5% oleh pemberi kerja dan 2,5% ditanggung oleh pekerja. Sementara untuk peserta pekerja mandiri, besaran iuran yang harus dibayarkan disesuaikan dengan penghasilan yang dilaporkan, sebagaimana diatur oleh kebijakan negara. (CNN Indonesia, 2024)

Peraturan Pemerintah tentang TAPERA mengatur berbagai aspek program tabungan perumahan rakyat. Peserta TAPERA adalah pekerja berusia minimal 20 tahun atau sudah menikah dengan penghasilan minimal upah minimum. Program ini banyak memiliki pro dan kontra apalagi menyangkut dengan pekerja dengan status menengah kebawah dengan pertimbangan berbagai potongan setiap bulannya, adapun utk pengeluaran potongan tiap bulan biasanya mencapai 10% dari gaji disetiap bulannya termasuk dengan potongan TAPERA. Karena secara tak langsung kebijakan ini menyamaratakan status sosial ekonomi semua pekerja di Indonesia. (CNN Indonesia, 2024)

Media sosial telah menjadi ruang utama bagi masyarakat untuk mengekspresikan opini, emosi, dan respon terhadap berbagai peristiwa secara cepat

dan terbuka. Salah satu akun yang memiliki pengaruh besar dalam membentuk dan mencerminkan opini publik di Indonesia adalah akun Instagram @lambeturah, yang dikenal luas karena kontennya yang memicu diskusi dan komentar dari berbagai kalangan. Meskipun informasi yang dibagikan oleh akun ini tidak selalu terverifikasi secara faktual, kolom komentarnya merepresentasikan dinamika sentimen publik yang otentik, spontan, dan beragam. Hal ini menjadikan @lambeturah sebagai sumber data yang menarik untuk dianalisis dalam studi sentimen, khususnya untuk menggambarkan bagaimana masyarakat merespons isu-isu sosial dan budaya secara real-time dalam konteks bahasa sehari-hari yang alami. Oleh karena itu, meskipun terdapat keterbatasan dalam hal validitas konten, pemanfaatan data dari akun ini tetap relevan karena fokus penelitian lebih mengarah pada analisis persepsi dan pola emosional pengguna media sosial.

Di masa teknologi informasi yang semakin pesat, terutama media masa, memberikan kemudahan bagi masyarakat untuk menyampaikan pendapat sentiment mereka secara langsung dan *real-time*. Untuk menghadapi kompleksitas data teks yang dihasilkan oleh pengguna Instagram terkait TAPERA, diperlukan pendekatan analisis yang canggih dan efektif. Salah satu pendekatan yang dapat digunakan adalah *word embedding*, teknik yang mengubah kata-kata menjadi representasi *numeric* dalam ruang *vector* multidimensi. *Word embedding*, seperti *Word2Vec* yang bertujuan untuk menghubungkan setiap kata dalam teks ke dalam ruang vector. Dimana katakata dengan makna mirip atau sering kali muncul bersama-sama akan memiliki representasi *vector* yang berdekatan satu sama lain. Representasi *vector* ini memungkinkan komputer untuk menggali dan memahami hubungan semantic antara kata-kata. Dengan memanfaatkan *Word2Vec*, analisis ini dapat menghasilkan *vector* yang merepresentasikan kata-katadalam bentuk numeric. Representasi *vector* bermanfaat dalam berbagai tugas pemrosesan bahasa alami, seperti analisis *sentiment*, klasifikasi teks, pencarian informasi, dan penerjemah mesin. (Fiqih Aulia Pradana, 2023)

Karena akan menghasilkan beberapa kategori dalam pengelompokan data, maka penelitan ini menggunakan algoritma klasifikasi *Naives Bayes*. Algoritma ini yang umum digunakan dalam analisis *sentiment* karena lebih cocok dalam

penelitian. Algoritma ini didasarkan pada teorema *Bayes* yang mengasumsikan independensi antara fitur-fitur input, terkait dalam konteks kenyataan, independensi ini seringkali tidak sepenuhnya terpenuhi. Meskipun demikian, *Naives Bayes* tetap menjadi pilihan yang popular karena kemampuannya dalam mengklasifikasikan teks ke dalam kategori *sentiment* yang berbeda seperti positif, negatif dan netral. Dan juga dengan kemungkinan data kurang seimbang maka dari itu ditambah menggunakan algoritma *SMOTE (Sythetic Minority Oversampling Technique)* guna berupaya untuk keseimbangan data yang dihasilkan terhadap data negatif dan positif. Karena dilihat dari sampling data yang tersedia bahwa data negatif lebih banyak dibanding data positif, sehingga penggunaan algoritma *SMOTE* perlu ditambahkan. (Normah dkk, 2022)

Dalam penelitian terdahulu mengenai *Pelabelan Otomatis Lexicon Vader* dan Klasifikasi *Naive Bayes* dalam menganalisis sentimen data ulasan PLN Mobile. Perhitungan digunakan untuk meberikan informasi terkait interaksi ulasan aplikasi dengan memperhitungkan presentasi positif, negatif dan netral. (Fajri Muhamad dkk, 2022) Selain itu penelitian terdahulu yang membahas mengenai Implementasi *Word2Vec* pada Analisis Sentimen Terhadap Ulasan Pengguna Aplikasi Tiktok Menggunakan Metode *Support Vector Machine* memberikan informasi terkait dengan analisis *sentiment* dengan opini publik melalui aplikasi tiktok untuk membuktikan keakuratan metode *Word2Vec* dengan topic tersebut. (Denny Ivan Rufai, 2024)

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka rumusan masalah yang dapat diambil, yaitu :

1. Bagaimana sentiment pengguna Instagram terhadap Tabungan Perumahan Rakyat (TAPERA) menggunakan metode *Word2Vec* dengan algoritma *Naives Bayes & SMOTE*?
2. Bagaimana akurasi keseimbangan yang dihasilkan dari penggunaan algoritma *SMOTE* mengenai TAPERA?

1.3 Batasan Penelitian

Terdapat 5 Batasan masalah dalam penelitian ini, yaitu :

1. Penelitian ini hanya fokus pada pengguna *platform* instagram.
2. Data yang digunakan adalah postingan komentar yang terkait dengan simpanan tapera pada akun @lambeturah
3. Pengumpulan data dilakukan dengan menggunakan *API* Instagram untuk mengakses data publik yang relevan dengan topik tapera
4. Keterbatasn dalam kemampuan algoritma *Naives Bayes* dalam mengatasi konteks yang kompleks terkait kata-kata kurang baku dalam konteks bahasa
5. Data yang digunakan terbatas pada rentang waktu tertentu selama tahun 2024 tepatnya tanggal 28 Mei terkait penelitian yang dilakukan

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk mencapai tujuan berikut:

1. Memperoleh hasil opini sesudah dan setelah menggunakan metode *Word2Vec* dengan algoritma *Naives Bayes & SMOTE* terhadap TAPERA di Instagram.
2. Memperoleh hasil akurasi dari klasifikasi analisis sentimen terhadap Tabungan Perumahan Rakyat menggunakan algoritma *SMOTE*.

1.5 Manfaat Penelitian

Berdasarkan tujuan penelitian di atas manfaat yang bisa di ambil dari penelitian ini yaitu:

1. Bagi Peneliti

   Sebagai bahan penyusunan tugas akhir, dan bentuk penerepan keilmuan yang telah diperoleh selama perkuliahan di Universitas Jember,

serta memberikan wawasan dan persepsi sentimen masyarakat terhadap simpanan TAPERA.

2. Bagi Akademis

   Menambahkan literatur dalam bidang analisis sentiment menggunakan algoritma *Naive Bayes & SMOTE* dengan metode *Word2Vec* terkait dengan simpanan TAPERA.

3. Bagi Masyarakat

   Hasil penelitian ini dapat memberikan masukan berharga bagi pihak terkait seperti pemerintah atau lembaga terkait, untuk meningkatkan atau menyesuaikan strategi komunikasi serta implementasi program TAPERA sesuai dengan persepsi dan respon masyarakat.

BAB 2. TINJAUAN TEORI

2.1 Penelitian Terdahulu

Terdapat berbagai penelitian terdahulu yang berpotensi dan dapat digunakan sebagai acuan dalam penelitian yang akan dilakukan. Berikut ini adalah hasil kajian penelitian terdahulu dari beberapa jurnal yang terdapat pada tabel 2.1.

Tabel 2.1 Tabel Penelitian Terdahulu

Judul	Hasil	GAP	Kontribusi
Analisis Sentimen Komentar Instagram Pada Program Kampus Merdeka Dengan Algoritma Naive Bayes Dan Decision Tree (Bayu Wicaksono & Cahyono Nuri, 2024)	Hasil analisis sentimen menggunakan algoritma Complement Naïve Bayes dan Decision Tree menunjukkan, sebelum SMOTE over-sampling, perbandingan data positif dan negatif adalah 35,06% berbanding 64,95%. Akurasi model Decision Tree mencapai 84% pada skenario 90:10, sementara Complement Naïve Bayes 81% pada skenario 80:20. Setelah penerapan SMOTE, akurasi Decision Tree meningkat 1% menjadi 85%, - Object Pelelitian yang berbeda yaitu program TAPERA dan Program Kampus Merdeka - Penggunaan Algoritma yang berbeda yaitu Decision Tree dan Word2Vec Sebagai acuan pandangan bentuk system yang akan dihasilkan nantinya, serta perhitungan dari perbandingan variable yang diambil sedangkan Complement Naïve Bayes meningkat 2% menjadi 83%.	- Object Pelelitian yang berbeda yaitu program TAPERA dan Program Kampus Merdeka - Penggunaan Algoritma yang berbeda yaitu Decision Tree dan Word2Vec	Sebagai acuan pandangan bentuk system yang akan dihasilkan nantinya, serta perhitungan dari perbandingan variable yang diambil
SMOTE : Potensi Dan Kekurangannya Pada Survei (Wijayanti et al., 2021)	Keunggulan metode SMOTE secara umum meliputi tidak menghilangkan informasi, mencegah overfitting, memperluas wilayah keputusan, dan meningkatkan akurasi prediksi pada kelas minoritas. Namun, kelemahannya mencakup overgeneralization yang dapat menyebabkan overlapping, serta kurang cocok digunakan pada kasus yang mempertimbangkan kepentingan fitur dan data dengan dimensi tinggi. Untuk mengatasi masalah tersebut, berbagai perkembangan dan modifikasi telah dilakukan pada metode ini. Meskipun demikian, SMOTE tetap menjadi pelopor dalam pengembangan teknik oversampling yang menggunakan data sintetis.	- Penggunaan SMOTE over sampling yang lebih umum dan banyak digunakan disbanding beberapa jenis SMOTE yang ada dalam jurnal - Penambahan metode SMOTE dalam Word2Vec	Penelitian ini menggambarkan penggunaan metode SMOTE dalam penelitian ini.

Judul	Hasil	GAP	Kontribusi
Implementasi Word2vec Pada Analisis Sentimen Terhadap Ulasan Pengguna Aplikasi Tiktok Menggunakan Metode Support Vector Machine (Denny Ivan, 2023)	Peneliti menggunakan model word2vec yaitu cbow (continuous bag of words) dan skip gram untuk membandingkan akurasi serta membuat klasifikasi teks dengan jumlah dataset 900 terbagi menjadi 300 negatif,300 positif dan 300 netral serta menerapkan metode Support Vector Machine (SVM) untuk mengklasifikasikan sentimen dari ulasan pengguna aplikasi tiktok. Metode penelitian ini melibatkan langkah langkah utama, yaitu pengumpulan dataset ulasan pengguna TikTok, preprocessing data, dan implementasi model Word2Vec. Setelah representasi vektor kata-kata berhasil dihasilkan, SVM diterapkan untuk melatih dan menguji model analisis sentimen. Hasil akurasi dari model cbow (continuous bag of words) 0.66 sedangkan hasil model skip gram 0.68. hasil akurasi cukup rendah namun untuk dapat meningkatkan hasil akurasi perlu meningkatkan jumlah dataset, fokus pada tahap preprocessing, yaitu memperbaiki kata yang tidak baku serta pengecekan kata yang lebih detail.	- Perbedaan penggunaan platform yang digunakan, yaitu Tiktok dan Instagram - Data tidak ada proses balancing dengan menggunakan SMOTE	Pengertian terkait penggunakan Word2Vec agar lebih memahami terkait system dan prosesnya.

2.2 Simpanan Tapera

TAPERA (Tabungan Perumahan Rakyat) adalah program yang ditujukan untuk membantu masyarakat Indonesia dalam memiliki rumah. Simpanan TAPERA merupakan bentuk tabungan yang diperuntukkan bagi pekerja dan masyarakat yang ingin memiliki rumah. Program ini diluncurkan oleh pemerintah untuk mendukung akses perumahan yang layak dan terjangkau.

TAPERA mengandalkan tiga sumber simpanan: iuran bulanan peserta yang sekitar 3% dari gaji, kontribusi pemerintah untuk meningkatkan dana, dan pendapatan dari investasi dana yang terkumpul. Manfaat TAPERA meliputi akses pembiayaan perumahan bagi peserta yang memenuhi syarat, kemungkinan subsidi dari pemerintah untuk cicilan, dan peningkatan kemandirian masyarakat dalam memiliki rumah tanpa tergantung pada pinjaman berat. data nasional adalah fasilitas yang menyimpan dan mengelola data dalam skala besar. (CNN Indonesia., 2024)

2.3 Analisis Sentimen

Analisis *sentiment* adalah proses komputasional untuk mengidentifikasi, mengekstraksi, dan mengevaluasi opini, perasaan, atau emosi yang terkandung dalam teks atau data lainnya. Tujuan utama dari analisis sentimen adalah untuk memahami sikap atau reaksi subjektif dari individu atau kelompok terhadap suatu topik, produk, layanan, kejadian, atau isu tertentu.

Metode analisis sentimen dapat melibatkan berbagai pendekatan dari bidang pemrosesan bahasa alami *(NLP),* statistik, dan pembelajaran mesin untuk mengklasifikasikan teks menjadi kategori sentimen seperti positif, negatif, atau netral. Analisis sentimen memiliki aplikasi yang luas, dari penelitian pasar dan manajemen merek hingga pengambilan keputusan berbasis data dan pemahaman opini publik di media sosial. (Bayu Wicaksono, Nuri Cahyono., 2024)

2.4 *Word Embedding*

Metode *Word embedding* adalah teknik dalam pemrosesan bahasa alami *(NLP)* untuk merepresentasikan kata-kata sebagai *vector* numerik dalam ruang multidimensional. Representasi ini memungkinkan komputer untuk memahami makna semantik kata-kata berdasarkan konteksnya dalam teks. (Bayu Wicaksono, Nuri Cahyono., 2024)

Secara matematis, model *neural network* untuk *Skip-gram* dengan *negative sampling* dapat direpresentasikan sebagai berikut:

$$\text{WordEmbedding}\ (w_i) = V_i \qquad (1)$$

Dimana:

w_i adalah kata ke I dalam korpus

V_i adalah vector representasi dari kata w_i

Dalam *Word2Vec*, ada dua model yang digunakan: *Continuous Bag of Words* (CBOW) dan *Skip-Gram*.

CBOW : $P\ (w_t | w_{t-n}, \dots, w_{t+n})$

Skip-Gram : $P\left(w_{t-n,\dots,w_{t+n}} \middle| w_t\right)$

Keterangan :

V : Jumlah total kata dalam kosakata.

vw : Vector representasi kata w.

v'w : Vector representasi kata w pada *layer output*.

P (w| wcontext) : Probabilitas kata w muncul dalam konteks kata *wcontext*.

Tujuan dari pelatihan adalah untuk mengoptimalkan *vector vw dan v'w* sehingga probabilitas. *P (w| wcontext)* dapat diprediksi dengan tepat berdasarkan konteks yang diberikan.

2.5 *Word2Vec*

Rumus komputasi utama dalam *Word2Vec* didasarkan pada fungsi objektif yang disebut sebagai *cost function*, yang dirancang untuk meminimalkan perbedaan antara produk komentar dari nilai *vector* representasi. *Cost Function (Objective Function)* :

$$T\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0} logp\left(w_{t+j}\middle|w_t\right) \quad (2)$$

$w_t =$ kata *center*

$w_{t+j} =$ kata setelah kata *center*

c = ukuran *training context*

Nilai *vector* dokumen atau dalam penelitian merupakan nilai *vector* komentar. Nilai *vector* dokumen didapatkan dengan menghitung nilai rata-rata *vector* dari semua kata di dalam satu topik. berikut persamaan untuk menghitung nilai *vector* dokumen. (Kurniawan, F. W., Maharani, W., 2020)

$$vec_{c,j} = \frac{1}{n_j}\sum_{i=1}^{m} w_i v(w_i) \quad (3)$$

$vec_{c,j} =$ *vector* dokumen j pada class c

$n_j =$ jumlah fitur dalam j

$v(wi) =$ *vector* kata dari fitur w

$w_i =$ nilai *vector* dari fitur w

Keuntungan *Word2Vec* :

1. *Word2Vec* menghasilkan representasi kata-kata dalam bentuk *vector* yang di olah dalam arti kemiripan karena mempertimbangkan dataset yang ada.
2. Lebih efektif dalam menganalisis dan ketepatan dalam bentuk persentase.

2.6 Algoritma *Naive Bayes Classifier*

Algoritma *Naive Bayes* adalah metode klasifikasi yang berdasarkan teorema *Bayes* dengan asumsi sederhana bahwa semua fitur (atau kata-kata dalam konteks analisis teks) adalah independen satu sama lain. Dalam konteks klasifikasi teks, *Naive Bayes* digunakan untuk memprediksi kelas dari sebuah dokumen atau teks berdasarkan kemunculan kata-kata dalam teks tersebut. (Domingos, P., & Pazzani, Michael,. 1997)

$$P(Y|X) = \frac{P(Y|X).P(X)}{P(Y)} \quad (4)$$

$(Y|X) =$ Kejadian

$P(X|Y) =$ *Probability* untuk X ketika Y benar

$P(Y|X) =$ *Probability* untuk Y ketika X benar

$P(X), P(Y) =$ *Probability independent* untuk X dan Y

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e - \frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2 \quad (5)$$

Langkah-langkah Algoritma Naive Bayes:

1. Pemilihan Model

   Pilih jenis model *Naive Bayes* yang sesuai dengan data, seperti *Naive Bayes Multinomial* untuk data teks dengan jumlah kata-kata, atau Naive Bayes *Gaussian* untuk data *biner*.

2. Pelatihan

   $P(X|Y) =$ *Probability* untuk X ketika Y benar

   $P(Y|X) =$ *Probability* untuk Y ketika X benar

Klasifikasi

$P(X), P(Y) =$ *Probability independent* untuk X dan Y

Kelebihan dan Keterbatasan:

Kelebihan:

1. Sederhana dan cepat dalam pelatihan dan klasifikasi.
2. Efektif dalam dataset dengan fitur yang besar.
3. Tahan terhadap *noise* dalam data.

Keterbatasan:

1. Asumsi *independensi* antar fitur sering tidak realistis dalam praktiknya.
2. Rentan terhadap *overfitting* jika data pelatihan terlalu sedikit.
3. Tidak mampu menangani hubungan kompleks antar fitur.

2.7 SMOTE

Data yang digunakan merupakan data *imbalance* atau data tidak seimbang. Kondisi tersebut apabila pada suatu dataset memiliki data yang sangat besar pada kelas mayoritas dibandingkan dengan kelas minoritas. Perbedaan tersebut menyebabkan model klasifikasi tidak dapat memperediksi dengan tepat.Untuk mengatasi permasalahan tersebut diperlukannya *balancing* data atau penyeimbangan data. Salah satu teknik yang sering digunakan yaitu *Syntetic Minority Over-sampling Technique (SMOTE)* metode *Oversampling.*

BAB 3. METODOLOGI PENELITIAN

3.1 Jenis Penelitian

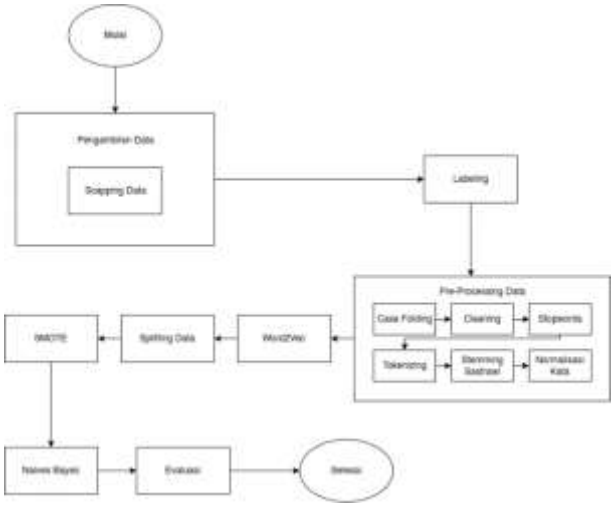
Penelitian kuantitatif merupakan jenis penilitian terstruktur dengan fokus pada pengukuran numerik, statistik, atau data terstruktur untuk memahami fenomena yang diteliti. Dalam analisis sentimen, penelitian kuantitatif digunakan untuk memproses sejumalh besar data dengan tujuan mengidentifikasi sentimen yang terkandung di dalamnya.

3.2 Tempat dan Waktu Penelitian

Penelitian dilakukan pada media sosial Instagram. Data yang digunakan dalam penelitian ini yakni sentimen terhadap potongan simpanan tapera yang diposting oleh akun *@lambeturah* pada tanggal 28 Mei 2024 yang masih sering menjadi topik pembahasan hangat hingga saat ini.

3.3 Tahapan Penelitian

Tahapan penelitian ini dibuat agar penelitian yang diperoleh tidak menyimpang dari tujuannya. Langkah-langkah pada tahapan penelitian yang dilakukan adalah sebagai berikut :



Gambar 3.1 Blok Diagram Tahapan Penelitian

3.4 Pengambilan Data

Data yang digunakan dalam penelitian ini adalah pada media sosial instagram, yakni sentimen terhadap potongan simpanan tapera pada tanggal 28 Mei 2024. Topik tersebut mulai muncul diakhir tahun 2023 hingga dibahas dalam postingan instagram oleh akun @lambeturah, jumlah *data testing* algoritma *Naïve Bayes* yang digunakan penulis yaitu 1000 dari total 20,5ribu data komentar. Dalam pengambilan data dari instagram kami menambang data (*crawling* data) menggunakan bantuan *API* Instagram dengan bantuan ekstension pada *Google Chrome* yaitu *IG Exporter & Scrapper.*

3.5 *Labelling Data*

Data yang telah diambil kemudian masuk pada tahapan *labeling* yang berguna untuk memberikan tanda terhadap sentiman positif dan negatif pada setiap data komentar yang diambil dengan mengambil data sampel sebanyak 1000 untuk diberi label melalui microsoft excel secara manual.

3.6 *Preprocessing Data*

3.6.1. Cleaning

Penghapusan simbol, karakter, hastag(#), username(@username), url(https://situs.com), email (nama@domain.com) untuk mengurangi noise pada kalimat atau element lain yang tidak memberikan kontribusi pada analisis sentiment pada data komentar yang telah masuk proses ini.

3.6.2. Casefolding

Kalimat akan diubah menjadi huruf kecil semua *(lowercase).* Dalam analisis sentimen pada data komentar Instagram, langkah ini membantu mengurangi variasi dalam kata-kata dengan huruf kapital atau huruf kecil, memastikan konsistensi dalam analisis.

3.6.3. Normalisasi

Kata akan diubah ke dalam bentuk baku, memperbaiki kata gaul dan kata berbahasa inggris. Proses Normalisasi kata hampir sama dengan proses *stemming,* akan tetapi proses stemming hanya menghapus imbuhan pada awalan dan akhiran kata. Normalisasi kata membantu proses analisis *sentiment* untuk mendapatkan data yang lebih baik.

3.6.4. Tokenizing

Tokenizing merupakan pengubahan dari kalimat atau teks menjadi beberapa kata. Proses *tokenizing* penting dikarenakan dapat membantu mesin memahami bahasa manusia, sehingga lebih mudah untuk dianalisis.

3.6.5. Stopword

Stopword merupakan pembuangan kata-kata umum yang sering muncul dalam teks dan biasanya tidak memberikan kontribusi signifikan terhadap makna atau *sentiment.*

3.6.6. Stemming

Penghapusan semua pengimbuhan awalan dan akhiran kata yang akan di ubah menjadi kata dasar. Penggunaan *library Sastrawi* dikarenakan kemampuan *stemming* dengan dukungan Bahasa Indonesia lebih baik dibandingkan dengan *library nltk.*

3.7 *Word2Vec*

Word2Vec pada penelitian ini digunakan untuk merepresentasikan teks menjadi *vector*. setelah itu nilai *vector* ada dibentuk jaringan yang saling berdekatan, kemudian pada setiap pasangan kata akan membangun suatu *corpus* dimana setiap bagian dimensi akan memiliki nilai yang besar. Sedangkan kegunaan *SMOTE* untuk balancing atau menyeimbangkan data ini untuk mengurangi akurasi yang kurang tepat. Untuk bagian *Naive Baiyes* digunakan untuk pengumpulan nilai suatu data yang kemudian dikelompokkan dalam dua

jenis negatif, dan positif. Tahapan dalam proses *Word2Vec* adalah sebagai berikut.

1. *Training Word2Vec.*

   Dalam penelitian ini, implementasi *Word2Vec* dilakukan dengan memanfaatkan model yang telah dilatih sebelumnya (*pre-trained*) menggunakan arsitektur *skip-gram*. Model *pre-trained* dipilih karena telah mempelajari representasi kata dari korpus data yang luas dan beragam, sehingga dapat memberikan hasil yang lebih optimal. Sebagai alternatif, pelatihan model *Word2Vec* juga dapat dilakukan secara mandiri pada dataset penelitian dengan mengonfigurasi beberapa parameter penting, yaitu ukuran *window*, dimensi *embedding*, dan *minimum count*

2. Representasi Teks

   Konversi teks menjadi representasi vektor menggunakan model *Word2Vec*. Misalnya, representasi dokumen bisa berupa rata-rata *vector* kata.

3.8 *Splitting Data*

Splitting dataset menggunakan proporsi 80:20 merupakan metode pemisahan data menjadi dua komponen utama, dimana 80% dialokasikan sebagai *training data* dan 20% sebagai *testing data*. Tahapan awal melibatkan proses *randomisasi* urutan dataset untuk memastikan penyebaran yang tidak memihak dan menghilangkan bias urutan.

Training data yang mencakup 80% dari keseluruhan dataset berfungsi dalam proses pembelajaran algoritma, sementara *testing data* dengan porsi 20% berperan untuk evaluasi kinerja model. Proses ini dapat diimplementasikan melalui berbagai *platform* pemrograman atau teknik manual. Proporsi 80:20 menjadi standar karena menyediakan volume data yang optimal untuk fase *training* sambil mempertahankan data yang *sufficient* untuk fase *testing* yang *reliable*.

3.9 SMOTE

SMOTE berguna untuk menggabungkan 2 perbandingan yang kurang seimbang menjadi lebih efektif. Dimulai dengan persiapan data *imbalanced*, program berguna untuk eksperimen sistematis terhadap parameter kunci untuk menemukan nilai optimal yang menghasilkan akurasi terbaik, setelah itu masuk proses *testing* dalam menentukan *sampling_strategy* (*'auto'*, *'minority'*, *custom dict*, dan *rasio float*) untuk mengontrol bagaimana *oversampling* dilakukan. Setelah menemukan konfigurasi optimal, program mengimplementasikan SMOTE dengan parameter terbaik dan membandingkan performa model sebelum dan sesudah SMOTE melalui *classification report* yang menunjukkan peningkatan *recall* pada kelas minoritas.

3.10 *Naive Bayes Classifier*

Menggunakan *Gaussian Naive Bayes* melakukan *training* data dengan menghitung statistik sederhana dari setiap fitur *numerik* berdasarkan kelasnya. selanjutnya data pelatihan dipisahkan menurut label kelas, kemudian untuk setiap fitur dalam setiap kelas, dihitung nilai rata-rata (*mean*) dan variansi (*variance*) yang diasumsikan mengikuti distribusi normal (*Gaussian*). Selain itu, model juga menghitung probabilitas awal (*prior*) untuk masing-masing kelas berdasarkan proporsinya dalam data.

3.11Evaluasi

Setelah melalui semua tahapan kemudian data akan menunjukan nilai akurasi yang berguna untuk evaluasi dari proses yang telah berjalan sepenuhnya. Evaluasi digunakan untuk memberikan sebesar besar nilai atau perbandingan antara data komentar negatif dan positif.

BAB 4. HASIL DAN PEMBAHASAN

4.1 Pengumpulan Data

Data peneliti diambil dari *Instagram* dengan kasus TAPERA. Dalam penelitian ini pengambilan data dari Instagram dilakukan dengan menambang data (*crawling data*) menggunakan bantuan *exstensi* dari *google chrome*. Dataset yang diambil adalah komentar dari media sosial Instagram dan dibatasi berjumlah 1000 data,

4.2 *Labelling* Data

Data yang telah dikumpulkan kemudian masuk pada proses *labeling* yang digunakan untuk memberikan nilai positif dan negatif dari berbagai komentar yang muncul di instagram terkait dengan topik Tapera. Data tersebut diberi label secara manual dengan acuan pada makna kata-kata yang mengandung unsur kebencian atau tidak. Berikut adalah hasil dari pelabelan manual dengan mengambil sampel 15 sentimen positif dan 5 sentimen negatif. Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.

Tabel 4.1 Tabel *Labelling* Data

Username	Text	Sentimen
igasan28	@tatiarin18 gak mau jadi biduan. Aja	positif
nugrohonatha	Udah cape dengan kondisi malah di potong lagi.. mau kalian ini apa sih para pejabat??? Cape tau cape.....	negatif
jevi_daihatsu.wonosobo	Yang kemarin ok gas mana nih	positif
roni_anggara24	Gpp lah	positif
jujuiyu3	@hary_agatha SETUJU é¦ƒæ§€	positif
coach_hendrik	@nurul.kr Serem gitu mbakk alasan nya... é¦ƒæ§€	negatif
mia_titans	Jeritan suara hati rakyat.é¦ƒæ§¶é¦ƒæ§¶é¦ƒæ§¶.	negatif

Username	Text	Sentimen
juniarizka	Potong sana potong sinii	negatif
jujuiyu3	Sesungguhnya p3njajah yang paling menyeramkan adalah MENJAJAH BANGSA SENDIRI é¦ƒæ§¶	negatif
joie_077	@rfauzia_ pura - pura tidak tahu é¦ƒî□•é¦ƒî□•	Positif

4.3 *Preprocessing Data*

Masuk pada bagian *pre-processing*, yaitu kondisi data mulai diproses untuk menjadi lebih akurat dan menghindari data komentar kurang *valid* ketika *coding* dijalankan, seperti simbol, atau emoji yang akan mempengaruhi *coding* yang berjalan sehingga diperlukan tahapan *pre-processing.* Adapun dibagian *pre-processing* dibagi menjadi 6 bagian proses secara berurutan sesuai dengan prosesnya, yang akan kami pecah sesuai dengan kondisi data yang komentar yang telah melewati tahapan *labeling.*

4.3.1. Casefolding dan Cleaning

Pada bagian ini kondisi data text komentar diproses untuk bersihkan dibagian huruf kapital & simbol.

Tabel 4.2 Tabel *Cleaning* dan *Casefolding*

Text	*Cleaning* dan *Casefolding*
@tatiarin18 gak mau jadi biduan. Aja	gak mau jadi biduan aja
Udah cape dengan kondisi malah di potong lagi.. mau kalian ini apa sih para pejabat??? Cape tau cape.....	udah cape dengan kondisi malah di potong lagi mau kalian ini apa sih para pejabat cape tau cape
Yang kemarin ok gas mana nih	yang kemarin ok gas mana nih
Gpp lah	gpp lah
@hary_agatha SETUJU ðŸ˜˜,	setuju

Terdapat perubahan sesuai dengan fungsi, yaitu terjadi perubahan pada bagian huruf kapital & simbol.

```
import re
def clean_ig(ig):
    ig = re.sub(r'@\w+', ' ', ig)
    ig = re.sub(r'#\w+', '', ig)
    ig = re.sub(r'http\S+|www\S+', '', ig)
    ig = re.sub(r'[^\w\s]', ' ', ig)
    ig = ig.strip()
    ig = ig.replace('²', ' ')
    ig = ig.lower()
    return ig
df['cleaning'] = df['text'].apply(lambda x: clean_ig(x))
```

Gambar 4.1 Kode Program *Cleaning* dan *Casefolding*

Dengan melakukan program tersebut menghasilkan sesuai dengan tampilan perubahan diatas. Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.

4.3.2. Normalisasi

Normalisasi kata digunakan untuk mengubah kata menjadi lebih baku, dengan mengubah kata gaul, kata *tren*, maupun singkatan.

Tabel 4.3 Tabel Normalisasi

Cleaning dan *Casefolding*	Normalisasi
gak mau jadi biduan aja	tidak mau jadi biduan saja
udah cape dengan kondisi malah di potong lagi mau kalian ini apa sih para pejabat cape tau cape	sudah capek dengan kondisi malah di potong lagi mau kalian ini apa sih para pejabat capek tau capek
yang kemarin ok gas mana nih	yang kemarin ok gas mana nih
gpp lah	tidak apa-apa lah
setuju	setuju

Berikut adalah tampilan hasil program ketika berhasil menggunakan Normalisasi pada data komentar pada *excel.*

```
kamus_data = pd.read_excel("kamuskatabaku.xlsx")
kamus_tidak_baku = dict(zip(kamus_data['tidak_baku'],
kamus_data['kata_baku']))
def normalisasi(comment):
    return ' '.join([kamus_tidak_baku[term] if term in
kamus_tidak_baku else term for term in comment.split()])

df['normalisasi'] = df['cleaning'].apply(normalisasi)
df = df[df['normalisasi'].str.strip().astype(bool)]
```

Gambar 4.2 Kode Program Normalisasi

Untuk kode program yang digunakan yaitu kamus kata baku dalam *excel* yang digunakan sebagai perubahan kata tidak baku menjadi kata yang baku. Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.

4.3.3. Tokenizing

Proses ini digunakan untuk melakukan pembagian sebuah kalimat menjadi sebuah kata satu persatu, yang berguna untuk proses *coding* programnya.

Tabel 4.4 Tabel Proses *Tokenizing*

Normalisasi	*Tokenizing*
tidak mau jadi biduan saja	['tidak', 'mau', 'jadi', 'biduan', 'saja']
sudah capek dengan kondisi malah di potong lagi mau kalian ini apa sih para pejabat capek tau capek	['sudah', 'capek', 'dengan', 'kondisi', 'malah', 'di', 'potong', 'lagi', 'mau', 'kalian', 'ini', 'apa', 'sih', 'para', 'pejabat', 'capek', 'tau', 'capek']
yang kemarin ok gas mana nih	['yang', 'kemarin', 'ok', 'gas', 'mana', 'nih']
tidak apa-apa lah	['tidak', 'apa-apa', 'lah']
setuju	['setuju']

Pemisahan kalimat menjadi sebuah kata satuan untuk masuk pada tahapan selanjutnya.

```
def tokenize(text):
  tokens = text.split()
  return tokens
df['tokenize'] = df['normalisasi'].apply(tokenize)
```

Gambar 4.3 Kode Program *Tokenizing*

Adapun kode program tersebut digunkan untuk membagi kalimat panjang menjadi kata persatuan dengna dipisahkan melalui tanda koma (,). Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.

4.3.4. Stopword

Dibagian ini data berupa kata-kata kemudian diubah untuk menghilangkan beberapa kata yang tidak masuk pada bagian pemrosesan atau kurang mengandung makna

Tabel 4.5 Tabel Hasil *Stopword*

Tokenizing	*Stopword*
['tidak', 'mau', 'jadi', 'biduan', 'saja']	['biduan']
['sudah', 'capek', 'dengan', 'kondisi', 'malah', 'di', 'potong', 'lagi', 'mau', 'kalian', 'ini', 'apa', 'sih', 'para', 'pejabat', 'capek', 'tau', 'capek']	['capek', 'kondisi', 'potong', 'sih', 'pejabat', 'capek', 'tau', 'capek']
['yang', 'kemarin', 'ok', 'gas', 'mana', 'nih']	['kemarin', 'ok', 'gas', 'nih']
['tidak', 'apa-apa', 'lah']	['apa-apa']
['setuju']	['setuju']

Kata diproses untuk menjadi lebih singkat lagi, membuang makna yang kurang berguna

```
def remove_stopwords(text):
  return [word for word in text if word not in stop_words]

df['stopword_removal'] = df['tokenize'].apply(lambda x:
remove_stopwords(x))
```

Gambar 4.4 Kode Program *Stopword*

Untuk bagian codingnya dipenelitian ini menggunakan *corpus* atau kamus tentang bahasa indonesia. Sehingga proses yang berjalan atau program yang dicoding tidak begitu banyak. Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.

4.3.5. Stemming

Untuk menyederhanakan makna dengan membuang kata berulang serta kata berimbuhan.

Tabel 4.6 Tabel Hasil *Stemming*

Stopword	*Stemming*
['biduan']	biduan
['capek', 'kondisi', 'potong', 'sih', 'pejabat', 'capek', 'tau', 'capek']	capek kondisi potong sih jabat capek tau capek
['kemarin', 'ok', 'gas', 'nih']	kemarin ok gas nih
['apa-apa']	apa
['setuju']	tuju

Untuk kode programnya sebagai berikut, menggunakan *library Sastrawi* dengan *StemmerFactory.*

```
factory = StemmerFactory()
stemmer = factory.create_stemmer()

def stemming(text):
  return [stemmer.stem(word) for word in text]

df['stemming'] = df['stopword_removal'].apply(lambda x: ' '.join(stemming(x)))
```

Gambar 4.5 Kode Program *Stemming*

4.4 *Word2Vec*

Pada bagian tahapan proses ini data yang telah melewati *pre-processing* kemudian diberi nilai *vector* pada setiap kata sesuai dengan *gensim* model yang telah tersedia dengan *library word2vec* model *skipgram.* Selanjutnya data *pre-processing* yang diolah akan memiliki nilai *vector.*

```
word2vec_model_skipgram = Word2Vec(vector_size=100, window=5, min_count=4, sg=1, workers=30, epochs=100)
word2vec_model_skipgram.build_vocab(df['stemming_list'], progress_per=10)
word2vec_model_skipgram.train(df['stemming_list'], total_examples=word2vec_model_skipgram.corpus_count, epochs=word2vec_model_skipgram.epochs)

import numpy as np
def get_document_vector(document):
    word_vectors = [word2vec_model_skipgram.wv[word] for word in document if word in word2vec_model_skipgram.wv]
    if word_vectors:
        return np.mean(word_vectors, axis=0)
    else:
        return np.zeros(word2vec_model_skipgram.vector_size)

document_vectors = [get_document_vector(doc) for doc in df['stemming_list']]
df_word2vec = pd.DataFrame({'skipgram': document_vectors})
df = pd.concat([df, df_word2vec], axis=1)
```

Gambar 4.6 Kode Program *Word2Vec*

Berikut merupakan *coding* program pada bagian tahapan *word2vec,* dengan menggunakan *gensim* serta model *skipgram.* Untuk output yang dimunculkan yaitu nilai *vector* pada kata-kata.

```
=== Informasi Model ===
Vector size: 100
Window size: 5
Min count: 4
Total words in vocabulary: 210
Training algorithm: 1
Epochs trained: 100
```

Gambar 4.7 Hasil Proses *Word2Vec*

Adapun untuk proses tambahan memperjelas program terkait kata yang mengandung *vector* adalah sebagai berikut. Ada 2 perhatian untuk melihat model kata dan setiap *vector* tersebut.

```
if word in model.wv:
    vector = model.wv[word]
    print(f"\nVector untuk kata '{word}':")
    print(f"Shape: {vector.shape}")
    print(f"First 10 dimensions: {vector[:10]}")
else:
    print(f"Kata '{word}' tidak ada dalam vocabulary")

try:
    similar_words = model.wv.most_similar("bagus", topn=10)
    print(f"\n=== Kata yang mirip dengan 'bagus' ===")
    for word, similarity in similar_words:
        print(f"{word}: {similarity:.4f}")
except KeyError:
    print("Kata 'bagus' tidak ditemukan dalam vocabulary")
```

Gambar 4.8 Kode Program Tambahan *Word2Vec*

Setiap bagian *word2vec* harus ditambahin beberapa *coding* program lainnya untuk memastikan bahwa program lebih *valid* dan output terlihat jelas. Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.

4.5 *Splitting Data*

Proses pembagian data test menjadi 2 yaitu 80% dan 20% digunakan untuk memberikan model lebih stabil dengan aturan yang telah ditentukan sebelumnya yaitu sebesar 80% dan 20%.

4.6 SMOTE

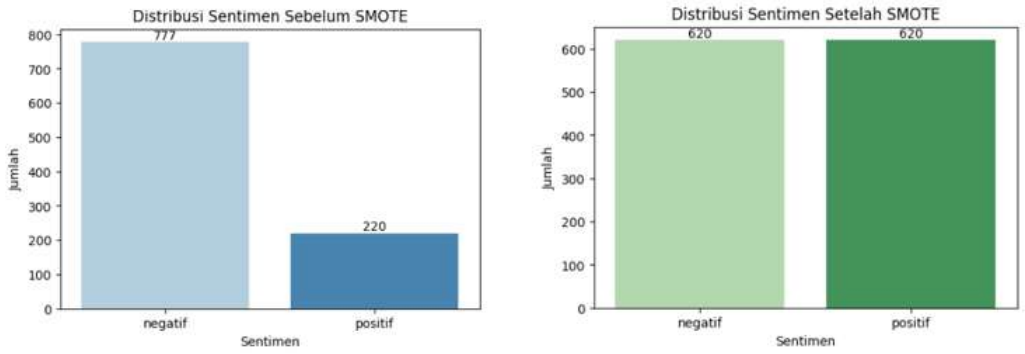
Proses pembagian data untuk mengurangi tingkat ketidakseimbangan antara data positif dan negatif, apapun berhitungan sampel telah penelitian ini coba, perbedaan jelas antara sesudah menggunakan *SMOTE* dan sebelum, adapun coding program akan saya tampilkan sebagai berikut serta hasil dari sebelum dan setelah menggunakan *SMOTE* dalam bentuk diagram.

```
sentiment_counts_before = data['sentimen'].value_counts()

plt.figure(figsize=(6, 4))
sns.barplot(x=sentiment_counts_before.index,
y=sentiment_counts_before.values, palette="Blues")
plt.title("Distribusi Sentimen Sebelum SMOTE")
plt.xlabel("Sentimen")
plt.ylabel("Jumlah")
for i, v in enumerate(sentiment_counts_before.values):
    plt.text(i, v + 1, str(v), ha='center', va='bottom')
plt.show()

smote = SMOTE(random_state=42)
X_train_smote,   y_train_smote   =   smote.fit_resample(X_train,
y_train)

resampled_data = pd.DataFrame(X_train_smote)
resampled_data['sentimen'] = y_train_smote

sentiment_counts_after                                          =
resampled_data['sentimen'].value_counts()

plt.figure(figsize=(6, 4))
sns.barplot(x=sentiment_counts_after.index,
y=sentiment_counts_after.values, palette="Greens")
plt.title("Distribusi Sentimen Setelah SMOTE")
plt.xlabel("Sentimen")
plt.ylabel("Jumlah")
for i, v in enumerate(sentiment_counts_after.values):
    plt.text(i, v + 1, str(v), ha='center', va='bottom')
plt.show()

resampled_data.to_csv("/content/drive/My             Drive/Colab
Notebooks/Hasil/Hasil_SMOTE.csv", index=False)
print("File Hasil_SMOTE.csv berhasil disimpan!")
```

Gambar 4.9 Kode Program SMOTE

Adapun grafik data sentiment sebelum dan setelah menggunakan SMOTE, yaitu kedua nilai memiliki perbedaan nilai yang signifikan yaitu sebelum berbanding 777 : 220 untuk negatif dan positif. Sedangkan setelah menggunakan SMOTE menjadi menjadi seimbang 620 : 620 menunjukan bahwa SMOTE

berfungsi sesuai dengan hasilnya yang membuat data seimbang. Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.



Gambar 4.10 Grafik Distribusi Sebelum dan Sesudah SMOTE

4.7 Impelmentasi SMOTE dan *Nave Bayes Classifier*

Klasifikasi *sentiment* yang digunakan untuk menghasilkan akurasi pada program ini yaitu sebagai berikut, setelah melalui berbagai tahapan sebelumnya. Adapun penelitian ini menggunakan *Naives Bayes* tipe *GaussianNB,* dengan *coding* program sebagai berikut.

```
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score, recall_score,
precision_score, f1_score
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import classification_report

classification = GaussianNB()
classification.fit(X_train_smote, y_train_smote)
```

Gambar 4.11 Kode Program *Naive Bayes Classifier*

Adapun prediksi akurasi dari data *sentiment* yaitu.

Classification Report:

	precision	recall	f1-score	support
negatif	0.83	0.54	0.65	157
positif	0.26	0.60	0.37	43
accuracy			0.55	200
macro avg	0.55	0.57	0.51	200
weighted avg	0.71	0.55	0.59	200

Gambar 4.12 Hasil Akurasi *Naive Bayes Classifier*

Dari Gambar 4.21 menghasilkan klasifikasi berupa metrik *accuracy* sebesar 0,71. sementara nilai *prescision* sebesar 0,55. untuk bagian metrik *recall* menunjukan nilai 0,57. adapun nilai dari *f1-score* sebesar 0,55.

Sebagian besar dari nilai hasil algoritma yang muncul memiliki tingkatan cukup, karena besaran diatas 0,50 sehingga dikatakan masih cukup baik atau stabil. Untuk hasil lebih lengkap dapat dilihat di halaman lampiran.

BAB 5. KESIMPULAN, KETERBATASAN, DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis sentimen dan pengujian model, kesimpulan yang didapatkan sebagai berikut.

1. Analisis sentimen model ini memiliki performa yang kurang dengan tingkat akurasi hanya 55%. Masalah utamanya adalah dataset yang tidak seimbang, dimana data negatif (157 sampel) jauh lebih banyak daripada data positif (43 sampel). Akibatnya, model lebih pandai memprediksi kelas negatif (precision 83%) tetapi sangat lemah dalam memprediksi kelas positif (precision hanya 26%). Hal ini berarti model sering salah menebak data negatif sebagai positif. Untuk memperbaikinya, perlu dilakukan penyeimbangan dataset, penyetelan ulang parameter model, dan pemilihan metrik evaluasi yang lebih tepat.

2. Penggunaan metode SMOTE dalam penelitian ini membuktikan keberhasilan untuk memperbaiki masalah data yang tidak seimbang pada analisis sentimen. Teknik ini mampu mentransformasi dataset yang semula sangat tidak proporsional (perbandingan 3,5:1) menjadi seimbang sempurna (perbandingan 1:1) dengan cara menciptakan data tambahan untuk kategori yang jumlahnya sedikit. Perbaikan keseimbangan data ini membawa manfaat besar untuk kinerja model pembelajaran mesin, antara lain meminimalkan kecenderungan hasil prediksi yang memihak, meningkatkan ketepatan analisis, serta menciptakan model yang lebih netral dalam mengenali sentimen positif maupun negatif. Ketika data sudah seimbang, model bisa memahami karakteristik kedua jenis sentimen dengan lebih optimal dan menghasilkan klasifikasi yang lebih terpercaya.

5.2 Saran

Saran yang dapat dilakukan untuk peneliti selanjutnya adalah meningkat kan dimensi pada *word2vec* untuk model *skip gram* yang mana bisa memungkinkan untuk meningkatkan akurasi, serta penambahan dataset yang lebih banyak agar dapat meningkatkan akurasi, dan fokus pada tahap *preprocessing*, yaitu memperbaiki kata yang tidak baku serta pengecekan kata yang lebih *detail*, serta membangun sistem klasifikasi teks yang lebih baik dan akurat dalam memberikan kategori *sentiment* positif dan negatif.

DAFTAR PUSTAKA

Asri, Y., Suliyanti, W. N., Kuswardani, D., & Fajri, M. (2022). Pelabelan Otomatis Lexicon Vader dan Klasifikasi Naive Bayes dalam menganalisis sentimen data ulasan PLN Mobile. Institut Teknologi PLN

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.

Church, K. W. (2017). Word2Vec. Natural Language Engineering, 23(1), 155-162.

CNN Indonesia. (2024, 29 Mei). Jejak Aturan Tapera di DPR, Disetujui Semua Fraksi Kini Tuai Polemik. Diakses Pada 7 Juli 2024, dari https://www.cnnindonesia.com/nasional/20240529100033-32 1103189/jejak-aturan-tapera-di-dpr-disetujui-semua-fraksi-kini-tuai polemik

CNN Indonesia. (2024, 28 Mei). PDIP Kritik Tapera Potong Gaji Karyawan: Untuk Sehari-hari Saja Sulit. Diakses pada 29 November 2024, dari https://www.cnnindonesia.com/nasional/20240528172819-32 1102981/pdip-kritik-tapera-potong-gaji-karyawan-untuk-sehari-hari-saja sulit

Domingos, P., & Pazzani, Michael. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. Machine learning, 29, 103-130. University Of California

Fiqih Aulia Pradana. (2023). Perbandingan Word Embedding Word2vec, Glove, Dan Fasttext Menggunakan Deep Learning Pada Ulasan Kondisi Pengguna Obat Kesehatan. Universitas Lampung.

Kurniawan, F. W., & Maharani, W. (2020). Analisis Sentimen Twitter Bahasa Indonesia dengan Word2Vec. eProceedings of Engineering, 7(2). Universitas Telkom

Normah, Bakhtiar Rifai, Satrio Vambudi, Rifki Maulana. (2022). Analisa Sentimen Perkembangan Vtuber Dengan Metode Support Vector Machine Berbasis Smote. Universitas Nusa Mandiri

Rifai, Denny Ivan. (2024). Implementasi Word2Vec Pada Analisis Sentimen Terhadap Ulasan Pengguna Aplikasi Tiktok Menggunakan Metode Support Vector Machine. Universitas Islam Sultan Agung Semarang.

Raizada, R. D., & Lee, Y. S. (2013). Smoothness without smoothing: why Gaussian naive Bayes is not naive for multi-subject searchlight studies. PloS one, 8(7), e69566.

Sujaini, H., & Putra, A. B. (2024). Analysis of language identification algorithms for regional Indonesian languages. IAES International Journal of Artificial Intelligence (IJ-AI), 13(2), 1741.

Wicaksono, Bayu., dan Nuri Cahyono. (2024). Analisis Sentimen Komentar Instagram Pada Program Kampus Merdeka Dengan Algoritma Naives Bayes dan Decision Tree. Universitas Amikom Yogyakarta.

Wijayanti, Ni Putu Yulika Trisna., Eka N Kencana., dan I Wayan Sumarjaya. (2021). SMOTE : Potensi dan Kekurangan Pada Survei. Univesitas Udayana.

Webb, G. I. (2017). Naïve bayes. In Encyclopedia of machine learning and data

mining (pp. 895-896). Springer, Boston, MA.
Wang, B., Wang, A., Chen, F., Wang, Y., & Kuo, C. C. J. (2019). Evaluating word embedding models: Methods and experimental results. APSIPA transactions on signal and information processing, 8, e19.

LAMPIRAN-LAMPIRAN

Lampiran 1.1 Dataset Mentah TAPERA : https://unej.id/datasettapera
Lampiran 2.1 Dataset Hasil *Preprocessing* : https://unej.id/datasethasilpreprocessingtapera
Lampiran 3.1 Kode Program Lengkap : https://unej.id/kodeprogramw2vdansmote