

**PENERAPAN METODE KLASIFIKASI *K-NEAREST NEIGHBOR* DAN
LEXICON BASED UNTUK ANALISIS SENTIMEN PUBLIK
TERHADAP PEMINDAHAN IBU KOTA NEGARA DENGAN
EKSTRAKSI FITUR *TF-IDF***

TUGAS AKHIR

Sebagai syarat untuk memperoleh gelar sarjana S-1 di Program Studi Informatika, Jurusan
Informatika, Fakultas Teknik Industri, Universitas Pembangunan Nasional “Veteran”

Yogyakarta



Disusun oleh:

Rifqi Maulana

123200128

PROGRAM STUDI INFORMATIKA

JURUSAN INFORMATIKA

FAKULTAS TEKNIK INDUSTRI

UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”

YOGYAKARTA

2025

HALAMAN PENGESAHAN PEMBIMBING

**PENERAPAN METODE KLASIFIKASI *K-NEAREST NEIGHBOR* DAN
LEXICON BASED UNTUK ANALISIS SENTIMEN PUBLIK TERHADAP
PEMINDAHAN IBU KOTA NEGARA DENGAN EKSTRAKSI FITUR *TF-IDF***

Disusun Oleh:

Rifqi Maulana

123200128

Telah diperiksa dan disetujui oleh pembimbing untuk disidangkan
pada tanggal :

Menyetujui,
Pembimbing

Rifki Indra Perwira, S.Kom., M.Eng.

NIP. 19830708 202121 1001

DAFTAR ISI

HALAMAN JUDUL	1
HALAMAN PENGESAHAN PEMBIMBING.....	2
DAFTAR ISI	3
DAFTAR GAMBAR	6
DAFTAR TABEL	7
DAFTAR LAMPIRAN	8
BAB 1	9
PENDAHULUAN.....	9
1.1 Latar Belakang	9
1.2 Rumusan Masalah	11
1.3 Batasan Masalah	11
1.4 Tujuan Penelitian.....	11
1.5 Manfaat Penelitian	12
1.6 Metodologi Penelitian dan Pengembangan Sistem.....	12
1.6.1 Metode Penelitian	12
1.6.2 Metode Pengembangan Sistem	13
1.7 Sistematika Penulisan	14
BAB II	15
TINJAUAN LITERATUR.....	15
2.1 Analisis Sentimen	15
2.2 Analisis <i>Term Frequency Inverse Document Frequency (TF-IDF)</i>	16
2.3 <i>X (Twitter)</i>	16
2.4 <i>Python</i>	17
2.5 <i>Web Scraping</i>	17
2.6 <i>SMOTE (Synthetic Minority Oversampling Technique)</i>	17
2.7 <i>Text Preprocessing</i>	18
2.7.1 <i>Text Cleaning</i>	18
2.7.2 <i>Tokenization</i>	18
2.7.3 <i>Pembakuan Kata</i>	18
2.7.4 <i>Negation Handling</i>	18
2.7.5 <i>Stopwords Removal</i>	18
2.7.6 <i>Stemming</i>	19

2.8 <i>Lexicon Based</i>	19
2.9 <i>Grid Search Cross Validation</i>	20
2.10 <i>K-Nearest Neighbors (KNN)</i>	21
2.11 Validasi dan Pengujian	22
2.12 Studi Pustaka.....	25
BAB III	28
METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM	28
3.1 Metodologi Penelitian	28
3.1.1 Pengumpulan Data	29
3.1.2 <i>Preprocessing</i>	30
3.1.3 <i>Sentiment Labelling</i>	38
3.1.4 Pembobotan <i>TF-IDF</i>	39
3.1.5 Penerapan Teknik <i>SMOTE</i>	44
3.1.6 Analisis Sentimen dengan Model <i>KNN</i>	50
3.1.7 Pembuatan Model Sentimen	52
3.1.8 Pengujian.....	53
3.2 Metode Pengembangan Sistem	53
3.2.1 <i>Requirements Analysis</i>	53
3.2.2 <i>System and Software Design</i>	54
3.2.3 <i>Implementation</i>	61
3.2.4 <i>System Testing</i>	61
BAB IV	62
HASIL DAN PEMBAHASAN	62
4.1 Implementasi	62
4.1.1 Implementasi Model Klasifikasi	62
4.1.2 Implementasi Pengembangan Sistem.....	70
4.2 Hasil	75
4.2.1 Proses <i>Training</i> Model.....	75
4.2.2 Proses Pengujian Model.....	77
4.2.3 Pengujian Pengembangan Sistem	79
4.3 Pembahasan.....	80

BAB V	81
KESIMPULAN DAN SARAN	81
5.1 Kesimpulan	81
5.2 Saran.....	81
DAFTAR PUSTAKA	82
LAMPIRAN.....	85

DAFTAR GAMBAR

Gambar 3.1 Metodologi Penelitian	28
Gambar 3.2 Flowchart Preprocessing	30
Gambar 3.3 Flowchart Text Cleaning	32
Gambar 3.4 Flowchart Pembakuan Kata	34
Gambar 3.5 Flowchart Negation Handling	35
Gambar 3.6 Flowchart Stopwords Removal	36
Gambar 3.7 Flowchart Stemming	37
Gambar 3.8 Flowchart TF-IDF	39
Gambar 3.9 Flowchart Jarak Nearest Neighbor SMOTE	44
Gambar 3.10 Flowchart Membuat Data Sintesis Baru SMOTE	46
Gambar 3.11 Flowchart Klasifikasi Model KNN	50
Gambar 3.12 Flowchart Pembuatan Model Sentimen	52
Gambar 3.13 Arsitektur Sistem	55
Gambar 3.14 Perancangan Proses	56
Gambar 3.15 Rancangan Halaman Beranda	57
Gambar 3.16 Rancangan Halaman Prediksi Sentimen	57
Gambar 3.17 Rancangan Halaman Dataset	58
Gambar 3.18 Rancangan Halaman Evaluasi	58
Gambar 3.19 Rancangan Halaman Sampel Tweet	59
Gambar 3.20 Rancangan Halaman Preprocessing	59
Gambar 3.21 Rancangan Halaman Sentimen Leksikon	60
 Gambar 4.1 Halaman Beranda	 70
Gambar 4.2 Halaman Prediksi Sentimen	71
Gambar 4.3 Tampilan Dataset Mentah	71
Gambar 4.4 Tampilan Dataset Setelah Preprocessing	72
Gambar 4.5 Tampilan Dataset Setelah Pelabelan Sentimen	72
Gambar 4.6 Halaman Evaluasi Model	73
Gambar 4.7 Halaman Sampel Tweet	73
Gambar 4.8 Halaman Preprocessing	74
Gambar 4.9 Halaman Sentimen Leksikon	74

DAFTAR TABEL

Tabel 2.1 Confusion Matrix	22
Tabel 2.2 Studi Pustaka	25
Tabel 2.3 Lanjutan Studi Pustaka	26
Tabel 2.4 Lanjutan Studi Pustaka	27
Tabel 3.1 Kata Kunci	29
Tabel 3.2 Contoh Hasil Pengumpulan Data	31
Tabel 3.3 Data Sebelum Text Cleaning	31
Tabel 3.4 Data Hasil Text Cleaning	32
Tabel 3.5 Data Sebelum Pembakuan Kata	33
Tabel 3.6 Data Hasil Pembakuan Kata	33
Tabel 3.7 Data Sebelum Negation Handling	34
Tabel 3.8 Data Hasil Negation Handling	35
Tabel 3.9 Data Sebelum Stopwords Removal	36
Tabel 3.10 Data Hasil Stopwords Removal	36
Tabel 3.11 Data Sebelum Stemming	37
Tabel 3.12 Data Hasil Stemming	37
Tabel 3.13 Data Sebelum Pelabelan Sentimen Leksikon	38
Tabel 3.14 Data Hasil Pelabelan Sentimen Leksikon	38
Tabel 3.15 Contoh Dokumen Perhitungan Term Frequency	40
Tabel 3.16 Hasil Term Frequency	40
Tabel 3.17 Lanjutan Hasil Term Frequency	41
Tabel 3.18 Hasil IDF	41
Tabel 3.19 Lanjutan Hasil IDF	42
Tabel 3.20 Hasil TF-IDF	43
Tabel 3.21 Hasil Jarak Nearest Neighbor SMOTE	45
Tabel 3.22 Hasil Data Sintesis SMOTE	47
Tabel 3.23 Dokumen Data Sintesis SMOTE	48
Tabel 3.24 Hasil Perubahan TF-IDF Setelah SMOTE	49
Tabel 3.25 Sampel Data Input KNN	51
Tabel 3.26 Nilai Jarak Data Uji Ke Setiap Data Latih	51
Tabel 3.27 Spesifikasi Hardware	54
Tabel 3.28 Software	54
Tabel 3.29 Rancangan Pengujian Black Box	61
Tabel 4.1 Persebaran Kelas	75
Tabel 4.2 Persebaran Kelas Setelah SMOTE	76
Tabel 4.3 Inisiasi Parameter	76
Tabel 4.4 Training KNN	76
Tabel 4.5 Pengujian Confusion Matrix	77
Tabel 4.6 Hasil Pengujian Black Box	79

DAFTAR LAMPIRAN

Lampiran 1. Sampel Data Pelabelan Manual.....	85
---	----

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Keputusan Presiden Indonesia untuk memindahkan ibu kota negara ke luar Pulau Jawa menjadi salah satu proyek strategis yang tertuang dalam Rencana Pembangunan Jangka Menengah Nasional Tahun Anggaran 2020-2024. Pada 26 Agustus 2019, Presiden yang pada saat itu sedang menjabat mengumumkan ibu kota negara baru ini akan dibangun tepatnya di Kabupaten Penajam Paser Utara dan sebagian Kutai Kartanegara, Kalimantan Timur (Hadi, 2020). Pemindahan ibu kota negara Indonesia merupakan salah satu proyek mega infrastruktur yang paling signifikan dalam beberapa dekade terakhir. Keputusan berani ini tidak hanya memicu perdebatan di kalangan para pengambil kebijakan, tetapi juga memicu beragam reaksi dari masyarakat luas.

Transisi menuju ibu kota baru negara Indonesia merupakan hal yang sangat sensitif sehingga banyak dibicarakan di media sosial, tidak terkecuali pada media sosial *X* (sebelumnya dikenal dengan nama *Twitter*). *X* merupakan media sosial yang seringkali menjadi pusat *trending* mengenai isu di dunia baik itu skala nasional maupun internasional yang dijadikan warganet sebagai media untuk menyuarakan opini terkait sentimen terhadap apa pun yang terkini diperbincangkan di jejaring sosial yang begitu kompleks (Sandi et al., 2023). Sosial media *X* menjadi salah satu platform yang sering digunakan oleh masyarakat Indonesia untuk mengekspresikan pendapat dan respons terhadap peristiwa-peristiwa penting seperti pemindahan ibu kota negara baru. Indonesia menempati posisi ke-5 di dunia sebagai pengguna media sosial *X* terbanyak menurut data dari *Statista* pada bulan Januari 2023. *Statista* merupakan salah satu situs *web* yang menyediakan data statistik yang dikenal di seluruh dunia.

Pada penelitian ini akan dilakukan analisa sentimen publik mengenai pemindahan ibu kota negara Indonesia di media sosial *X*. Teknik yang akan digunakan adalah *web scraping* untuk mengumpulkan data teks dari media sosial *X*. Penelitian ini akan dapat mengakses data teks yang mencakup berbagai macam opini, komentar, dan persepsi dari pengguna *X* terkait topik pemindahan ibu kota negara. Hal ini akan memungkinkan untuk didatakannya isi *dataset* dalam bentuk teks dan memadai untuk dilakukan analisis sentimen menggunakan metode klasifikasi *K-Nearest Neighbors (KNN)* dengan pelabelan sentimen *Lexicon Based*.

Metode *KNN* memberikan hasil yang kurang baik pada proses klasifikasi data jika terdapat *noise* atau informasi tambahan yang tidak berarti (Pratomo et al., 2021), metode tersebut juga sensitif terhadap data tidak seimbang (*imbalance data*) yang dapat menyebabkan kesalahan klasifikasi pada kelas minoritas (Pramayasa et al., 2023). Oleh karena itu, terdapat beberapa penelitian yang menyebutkan bahwa performansi pada metode *machine learning* dapat menghasilkan performa yang baik ketika dikombinasikan dengan ekstraksi fitur yang tepat (Pratomo et al., 2021). Teknik *resampling* juga dapat menjadi salah satu solusi dalam menangani kasus data yang tidak seimbang (Pramayasa et al., 2023). Penggabungan tersebut menawarkan pendekatan yang memadai dalam memproses dan menganalisis data teks untuk mengekstraksi informasi sentimen. Penerapan metode ini membuat kita dapat mengidentifikasi pandangan, sikap, dan respon yang diberikan publik terkait dengan topik pemindahan ibu kota negara tersebut.

Berdasarkan paparan sebelumnya, beberapa solusi yang digunakan adalah metode klasifikasi *K-Nearest Neighbors* dengan pelabelan sentimen berbasis *lexicon*, penyeimbang kelas dengan *SMOTE*, serta ekstraksi fitur *TF-IDF*. Metode *Lexicon Based* merupakan metode yang sederhana, layak, dan praktis untuk analisis sentimen dari data media sosial termasuk data kuesioner, data *Twitter*, dan data *Facebook* yang berupa opini suatu entitas tentang isu atau produk (Matulatuwa et al., 2017). *Lexicon Based* berperan sebagai pemberi label pada setiap kalimat opini yang ada pada *dataset* agar bisa digunakan sebagai *input* pelatihan model klasifikasi. Tahapan awal pada penelitian ini adalah *data preprocessing* bertujuan agar data yang nanti diproses lebih terstruktur, selanjutnya proses pelabelan dan ekstraksi fitur menggunakan metode berbasis *lexicon* dan *TF-IDF*, tidak lupa juga penyeimbangan kelas dengan *SMOTE*, lalu terakhir yaitu tahap klasifikasi. Beberapa teknologi yang telah disebutkan sebelumnya dipilih karena sudah dilakukannya peninjauan pustaka dari beberapa metode sejenis sebelumnya. Penelitian yang dilakukan oleh Setyo Adji Pratomo, dkk pada tahun 2021, dengan judul “Analisis Sentimen Pengaruh Kombinasi Ekstraksi Fitur TF-IDF dan Lexicon Pada Ulasan Film Menggunakan Metode KNN” berkesimpulan bahwa penggabungan fitur ekstraksi *TF-IDF* dengan leksikon *SentiWordNet* memiliki hasil akurasi 73.31%. Lalu, penelitian yang dilakukan oleh Azhar pada tahun 2018, dengan judul “Analisis Kinerja Algoritma Naïve Bayes dan K-Nearest Neighbor Pada Sentimen Analisis dengan Pendekatan Lexicon di Media Twitter”. *Dataset* yang digunakan merupakan data dari sosial media *X (Twitter)* dengan menggunakan *Twitter API*. Proses *Natural Language Processing* yang digunakan adalah *case folding*, *filtering*, *tokenizing*, normalisasi, *stopwords*, dan *stemming*. Hasilnya nilai *KNN* pada $k=5$ dengan tingkat akurasi mencapai 77%. Penelitian yang dilakukan oleh Muhammad Rayhan Elfansyah, dkk pada tahun 2024 dengan judul “Perbandingan Metode K-Nearest Neighbor (KNN) Dan Naïve Bayes Terhadap Analisis Sentimen Pada Pengguna E-Wallet Aplikasi Dana Menggunakan Fitur Ekstraksi TF-IDF” berkesimpulan bahwa metode *KNN* dan *Naïve Bayes* memiliki akurasi yang berbeda berdasarkan sumber label data. Pada data yang diberi label model *lexicon*, akurasi *KNN* mencapai 78% dan *Naïve Bayes* 74%. Berdasarkan penelitian yang dilakukan oleh Ahmad Hilman Dani, dkk pada tahun 2024 dengan judul “Studi Performa TF-IDF dan Word2Vec Pada Analisis Sentimen Cyberbullying” berkesimpulan bahwa metode *TF-IDF* memberikan performa terbaik dalam analisis sentimen dengan akurasi, *precision*, *recall*, dan *f1-score* yang lebih tinggi dibandingkan dengan model berbasis *Word2Vec* contohnya *CBOW* dan *Skip-gram*. Penelitian yang dilakukan oleh I Komang Dharmendra, dkk pada tahun 2024 yang berjudul “Evaluasi Efektivitas SMOTE dan Random Under Sampling Pada Klasifikasi Emosi Tweet” juga menyebutkan bahwa penerapan metode *oversampling* yang pada penelitian tersebut menggunakan *SMOTE* secara keseluruhan lebih efektif dalam meningkatkan performa model dibandingkan dengan metode *undersampling* atau pada kasus penelitian tersebut menggunakan *Random Under Sampling (RUS)*. *SMOTE* berhasil meningkatkan akurasi, presisi, *recall*, dan *f1-score* pada model klasifikasi. Sebaliknya, *RUS* cenderung menghasilkan performa yang lebih rendah karena pengurangan sampel dari kelas mayoritas yang dapat menghilangkan informasi penting.

Berdasarkan latar belakang yang telah dipaparkan, diharapkan penelitian ini dapat memberikan kontribusi dalam memahami bagaimana mengoptimalkan metode klasifikasi *K-Nearest Neighbors* dengan melakukan beberapa penyesuaian yang tepat dari mulai *preprocessing* sampai ke tahap klasifikasi sentimen untuk memahami reaksi publik, khususnya warganet di *X (Twitter)*, terhadap pemindahan ibu kota negara Indonesia, apakah lebih cenderung positif, netral, atau negatif. Hal ini dilakukan dengan mengintegrasikan data teks yang telah dibersihkan untuk diproses lebih lanjut oleh teknik pelabelan sentimen berbasis *lexicon* yang telah dikustomisasi, serta mencari nilai *k* yang optimal sebagai variabel penting dalam pembuatan model klasifikasi *KNN*, sebelum itu dilakukan penyeimbangan kelas dengan *SMOTE*, serta ekstraksi fitur *TF-IDF* untuk menghasilkan model analisis sentimen dengan akurasi tinggi. Selain itu, diharapkan penelitian ini dapat memberikan kontribusi dalam memahami bagaimana tanggapan serta dinamika sosial yang muncul saat terjadi perubahan besar, seperti pemindahan ibu kota. Hasil penelitian ini diharapkan dapat menjadi referensi bagi penelitian serupa di masa depan.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang sebelumnya, masalah yang dapat dirumuskan mencakup bagaimana tingkat akurasi dari analisis sentimen publik terkait pemindahan ibu kota negara Indonesia dengan model berbasis *KNN*.

1.3 Batasan Masalah

Agar masalah yang diteliti menghasilkan sasaran yang jelas, maka dibuatlah batasan masalah untuk menghindari adanya perluasan pembahasan ke depannya sebagai berikut:

1. Data penelitian yang digunakan merupakan data yang dihasilkan dari *web scraping* pada unggahan teks di media sosial *X (Twitter)*, mengenai sentimen publik atas keputusan Presiden terkait pemindahan ibu kota negara.
2. Dataset yang digunakan dalam penelitian berjumlah total 5341 baris unggahan teks memakai format *.csv*.
3. Kategori sentimen dibagi menjadi 3 yaitu positif, negatif, serta netral.
4. Leksikon yang digunakan adalah leksikon berbahasa Indonesia yaitu *InSet (Indonesia Sentiment Lexicon)* dari literatur yang disusun oleh Fajri Koto dan Gemala Y. Rahmanningtyas pada tahun 2017 dengan judul “InSet Lexicon: Evaluation of a Word List for Indonesian Sentiment Analysis in Microblogs”. Leksikon *InSet* terdiri atas 3,609 kata positif dan 6,609 kata negatif dengan bobot antara -5 sampai +5.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah mengetahui nilai akurasi dari model analisis sentimen berbasis *KNN* berdasarkan unggahan teks di media sosial *X (Twitter)* mengenai keputusan Presiden terkait pemindahan ibu kota negara.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat bermanfaat untuk memberikan kontribusi dalam pengembangan metodologi analisis sentimen berbasis teks, khususnya dalam penerapan algoritma klasifikasi *K-Nearest Neighbors* dengan pelabelan sentimen berbasis *lexicon* dalam konteks bahasa Indonesia. Penelitian ini juga diharapkan tidak hanya bermanfaat secara ilmiah sebagai referensi bagi penelitian-penelitian selanjutnya yang menggunakan pendekatan serupa, penelitian ini juga mendukung pengembangan algoritma *Natural Language Processing (NLP)* untuk teks berbahasa Indonesia. Selain itu, penelitian ini juga merupakan bagian dari pemenuhan salah satu syarat kelulusan strata satu (S1) Program Studi Informatika Fakultas Teknik Industri.

1.6 Metodologi Penelitian dan Pengembangan Sistem

Metode penelitian ini menggunakan metode penelitian kuantitatif. Metode penelitian kuantitatif merupakan penelitian empiris dimana data dalam bentuk sesuatu yang dapat dihitung atau angka (Punch, 1988). Berikut merupakan tahapan-tahapan penelitian yang dilakukan:

1.6.1 Metode Penelitian

1. Pengumpulan Data

Data yang akan digunakan dalam penelitian ini merupakan data yang diperoleh dari proses *web scraping* dari media sosial *X (Twitter)*. Data yang telah didapatkan akan disimpan dalam format *.csv*.

2. Preprocessing Data

Text Preprocessing memiliki tujuan untuk mengubah bentuk data yang tidak terstruktur menjadi data terstruktur. Pada tahapan ini, akan dilakukan *preprocessing* data mencakup *case folding*, *text cleaning*, pembakuan kata dengan kamus slang atau tidak baku, *negation handling*, *stopwords removal*, dan *stemming*.

3. Sentiment Labelling

Data yang telah melalui tahap *preprocessing*, kemudian dilakukan pelabelan sentimen terhadap data tersebut yang mencakup 3 kelas sentimen yaitu positif, negatif, dan netral. Pelabelan sentimen ini dilakukan menggunakan kamus *Lexicon*.

4. Ekstraksi Fitur

Tahapan ini akan membobotkan kata menggunakan metode *TF-IDF* untuk mengubah data yang sebelumnya berbentuk teks atau *string* ke bentuk matriks numerik.

5. Resampling

Tahapan ini melakukan teknik *oversampling* dengan salah satu metodenya yaitu *SMOTE* berfungsi sebagai penyeimbang kelas.

6. Klasifikasi *KNN*

Pembuatan model *KNN* dan menguji performanya untuk mengetahui hasil analisis sentimen menggunakan salah satu *python library* yaitu *scikit-learn*.

7. Pengujian

Tahapan ini melakukan pengujian kinerja model yang telah dikembangkan dengan menggunakan *confusion matrix* untuk menemukan nilai akurasi, presisi, dan *recall*.

1.6.2 Metode Pengembangan Sistem

Metode *Waterfall* merupakan salah satu model *SDLC* (*Software Development Life Cycle*) yang sering digunakan dalam pengembangan sistem informasi atau perangkat lunak. Metode ini menggunakan pendekatan sistematis dan berurutan. Tahapan dalam model ini dimulai dari tahap perencanaan hingga tahap pengelolaan (*maintenance*) dan dilakukan secara bertahap (Abdul Wahid, 2020). Tahapan metode *waterfall* adalah sebagai berikut:

1. *Requirements Analysis and Definition*

Merupakan tahapan awal yang melibatkan identifikasi dan pemahaman yang mendalam terhadap kebutuhan. Tujuan utamanya yaitu mengumpulkan persyaratan fungsional dan non-fungsional yang nantinya akan menjadi dasar dari pengembangan sistem.

2. *System and Software Design*

Tahapan perancangan sistem ini mengalokasikan kebutuhan-kebutuhan sistem pada perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan.

3. *Implementation*

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program.

4. *System Testing*

Merupakan tahap pengujian terhadap sistem yang telah dibuat yang bertujuan untuk mengetahui apakah sistem yang dibuat udah siap digunakan atau belum.

1.7 Sistematika Penulisan

Sistematika penulisan pada penelitian ini adalah sebagai berikut:

- | | |
|---------|---|
| BAB I | PENDAHULUAN
Pada bagian pendahuluan membahas mengenai latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, serta sistematika penulisan. |
| BAB II | TINJAUAN PUSTAKA
Tinjauan pustaka merupakan bagian yang memuat mengenai dasar teori yang digunakan untuk analisis serta perancangan sistem dan juga implementasi pada penelitian ini. Selain itu juga digunakan sebagai bahan referensi serta pondasi untuk memperkuat argumentasi pada penelitian ini. |
| BAB III | METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM
Pada bab ini membahas mengenai metodologi penelitian, analisis sistem dan perancangan sistem analisis sentimen. |
| BAB IV | HASIL, PENGUJIAN, DAN PEMBAHASAN
Bab ini menyajikan hasil dari penelitian yang berisi hasil implementasi dari perancangan yang telah dibuat pada bab sebelumnya. Selain itu berisi pengujian terhadap hasil penelitian beserta pembahasannya. |
| BAB V | KESIMPULAN DAN SARAN
Bab ini berisi kesimpulan dari hasil penelitian serta saran yang diajukan oleh penulis untuk pengembangan pada penelitian selanjutnya. |

BAB II

TINJAUAN LITERATUR

2.1 Analisis Sentimen

Analisis sentimen adalah bidang studi yang menganalisis pendapat, sentimen, evaluasi, penilaian, sikap dan emosi seseorang terhadap sebuah produk, organisasi, individu, masalah, peristiwa atau topik (Liu, 2012). Tujuan dari analisis sentimen yaitu untuk memahami opini, perasaan, serta pandangan yang terkandung pada teks atau data unstruktural lainnya. Pengaruh dan manfaat dari analisis sentimen menyebabkan penelitian mengenai analisis sentimen berkembang pesat, serta kurang lebih 20-30 perusahaan di Amerika berfokus pada layanan analisis sentimen (Liu, 2012). Manfaat sentimen analisis dalam dunia usaha antara lain untuk melakukan pemantauan terhadap suatu produk. Secara cepat dapat digunakan sebagai alat bantu untuk melihat respon masyarakat terhadap suatu produk sehingga dapat diambil langkah strategis berikutnya. Garis besar analisis sentimen itu sendiri bertujuan untuk mengekstrak atribut dan komponen dari beberapa komentar yang ada di media sosial dan sehingga dapat menentukan beberapa kelas positif, negatif dan netral (Permatasari et al., 2021).

Pada umumnya, sentimen analisis merupakan klasifikasi tetapi kenyataannya tidak semudah proses kualifikasi biasa karena terkait penggunaan bahasa, dimana terdapat ambiguitas dalam penggunaan kata serta perkembangan bahasa itu sendiri.

Menurut Liu (2012), analisis sentimen memiliki beberapa tahap untuk melakukan analisis sentimen, yaitu:

1. Level Dokumen

Level dokumen menganalisis satu dokumen penuh dan mengklasifikasikan dokumen tersebut memiliki sentimen positif atau negatif. Level analisis ini berasumsi bahwa keseluruhan dokumen hanya berisi opini tentang satu entitas saja. Level analisis ini tidak cocok diterapkan pada dokumen yang membandingkan lebih dari satu entitas.

2. Level Kalimat

Level kalimat menganalisis satu kalimat dan menentukan tiap kalimat sentimen bernilai positif, netral, atau negatif. Sentimen netral berarti kalimat tersebut bukan opini.

3. Level Aspek

Level aspek tidak melakukan analisis pada konstruksi bahasa (dokumen, paragraf, kalimat, klausa, atau frasa) melainkan melakukan langsung pada opini itu sendiri. Hal ini didasari bahwa opini terdiri dari sentimen (positif dan negatif) dan target dari opini tersebut. Tujuan level analisis ini adalah untuk menemukan sentimen entitas pada tiap aspek yang dibahas.

2.2 Analisis *Term Frequency Inverse Document Frequency (TF-IDF)*

TF-IDF atau *Term Frequency Inverse Document Frequency* merupakan metode pembobotan dengan menggabungkan metode *TF* dan *IDF*, metode ini memberikan bobot hubungan suatu kata terhadap dokumen (Wahyuni et al., 2017).

Proses frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. Sehingga bobot hubungan antara sebuah kata dan sebuah dokumen akan tinggi apabila frekuensi kata tersebut tinggi di dalam dokumen dan frekuensi keseluruhan dokumen yang mengandung kata tersebut yang rendah pada kumpulan dokumen.

Nilai *TF* dapat dihitung dengan rumus:

$$TF = \frac{\text{jumlah kata terpilih}}{\text{jumlah kata}} \dots\dots\dots(2.1)$$

Nilai *IDF* dapat dihitung dengan rumus:

$$IDF = \frac{\text{jumlah dokumen}}{\text{jumlah frekuensi kata terpilih}} \dots\dots\dots(2.2)$$

Nilai *TF-IDF*:

$$TFIDF = TF \times IDF \dots\dots\dots(2.3)$$

2.3 *X (Twitter)*

X adalah sebuah situs *web* yang dimiliki dan dioperasikan oleh Twitter Inc., yang menawarkan jaringan sosial berupa *microblog* sehingga memungkinkan penggunaanya untuk mengirim dan membaca pesan dengan sebutan *tweet* atau kicauan (Akbar et al., 2013).

Microblog adalah jenis alat komunikasi daring dengan manfaat agar pengguna dapat memperbarui status tentang mereka yang sedang memikirkan dan melakukan sesuatu, apa pendapat mereka tentang suatu objek atau fenomena tertentu. *Tweet* atau kicauan adalah teks tulisan hingga 140 karakter (atau lebih jika berlangganan fitur khusus pada platform tersebut) yang ditampilkan pada halaman profil pengguna. *Tweet* bisa dilihat secara publik atau dapat dibatasi pengiriman pesan ke daftar pengguna lain tertentu saja. Pengguna dapat melihat *tweet* pengguna lain yang dikenal dengan sebutan pengikut atau *followers* (Ramadhon, 2020).

2.4 Python

Python adalah bahasa pemrograman tingkat tinggi yang ditafsirkan, berorientasi objek dengan semantik dinamis. Struktur data bawaan tingkat tinggi, dikombinasikan dengan pengetikan dan pengikatan dinamis membuatnya sangat menarik untuk *Rapid Application Development*, serta digunakan sebagai bahasa skrip atau lem untuk menghubungkan komponen yang ada bersama-sama.

Sintaks *python* yang sederhana dan mudah dipelajari menekankan keterbacaan dan karenanya mengurangi biaya pemeliharaan program. *Python* mendukung modul dan paket yang mendorong modularitas program dan penggunaan kembali kode. Penerjemah *python* dan perpustakaan standar yang luas tersedia dalam bentuk sumber atau biner tanpa biaya untuk semua *platform* utama, dan dapat didistribusikan secara bebas (*Python – Wikipedia, 2017*).

2.5 Web Scraping

Web scraping merupakan sebuah teknik untuk mendapatkan informasi dari situs web secara otomatis tanpa harus menyalinnya secara manual (Ayani et al., 2019). Teknik *web scraping* memungkinkan konten utama yang terdapat pada situs dapat diekstraksi, dihimpun, kemudian dapat diproses. *Web scraping* akan melakukan ekstraksi data pada *World Wide Web* (*www*), kemudian data yang didapat akan disimpan pada *file system* atau basis data yang nantinya bisa diambil kembali atau dianalisis (Setiawan et al., 2020).

Cara kerja *web scraping* adalah dengan mengakses halaman pada *web*, menentukan data yang dalam halaman tersebut, melakukan ekstraksi, dan transformasi bila diperlukan, kemudian menyimpan data tersebut menjadi dataset terstruktur (Boeing, 2016).

2.6 SMOTE (Synthetic Minority Oversampling Technique)

Metode *SMOTE* (*Synthetic Minority Oversampling Technique*) adalah metode populer yang digunakan untuk menangani ketidakseimbangan kelas (*imbalanced class*), teknik ini mensintesis sampel baru dari kelas minoritas untuk menyeimbangkan kumpulan data dengan melakukan *resampling* kelas minoritas (Siringoringo, 2018). Contoh proses *SMOTE* (Barus, 2022):

1. Ambil sampel acak contohnya P1
2. Kemudian terapkan algoritma *KNN* pada P1
3. Ambil jarak tetangga terdekat P1, contohnya P2
4. Lalu *generate* data baru dengan persamaan:
$$P1' = P1 + rand(0,1) \times (P2 - P1) \dots \dots \dots (2.4)$$
5. Ulang terus proses tersebut hingga jumlah data minoritas sebanyak data mayoritas

2.7 Text Preprocessing

Tahapan *preprocessing* merupakan tahapan awal untuk mempersiapkan dokumen agar lebih mudah untuk diproses, tahapan *preprocessing* sebelum proses klasterisasi meliputi *cleansing*, *case folding*, *tokenizing*, *filtering*, dan *stemming* (Amalia et al., 2018). Berdasarkan hal tersebut, mengubah data yang sebelumnya tidak terstruktur memerlukan proses pengubahan menjadi data yang terstruktur untuk diproses pada langkah berikutnya. Berikut adalah beberapa tahapan dari *preprocessing*:

2.7.1 Text Cleaning

Text cleaning atau *cleansing* merupakan proses pembersihan data yang bertujuan untuk menghilangkan elemen-elemen yang tidak dibutuhkan pada sebuah dokumen. *Cleansing* bertujuan untuk memperbaiki kualitas data. Pada penelitian ini tahapan *cleansing* berfungsi untuk menghilangkan *mention*, *hashtag*, *url* dan *uri*, tanda baca, *emoji*, menghilangkan angka-angka, serta *case folding* atau mengubah huruf besar ke kecil.

2.7.2 Tokenization

Tokenization atau tokenisasi merupakan tahapan pemisahan teks menjadi potongan-potongan berupa huruf, kata, atau kalimat menjadi kata yang tidak terhubung. Data teks yang masuk pada tahap *tokenization* akan diubah menjadi potongan-potongan kata. Pada umumnya, karakter spasi membedakan atau mengidentifikasi setiap kata satu sama lain. Oleh karena itu, proses *tokenization* pada dokumen bergantung pada karakter spasi. Sebagai contoh, tokenisasi pada kalimat “ibu kota pindah” menghasilkan tiga *token*, yaitu “ibu”, “kota”, dan “pindah”.

2.7.3 Pembakuan Kata

Proses ini mengganti kata slang atau gaul menjadi kata baku dengan menggunakan kamus kata slang yang didapatkan dari beberapa sumber di internet agar lebih mudah terbaca oleh kamus *lexicon* saat dilakukan pelabelan sentimen.

2.7.4 Negation Handling

proses *negation handling* mengubah bentuk kata yang memiliki makna yang sama tanpa kata negasi seperti “tidak” dan “belum”. Contohnya seperti “tidak bagus” yang maknanya sama dengan “jelek”, atau “tidak susah” yang sama maknanya dengan “mudah”.

2.7.5 Stopwords Removal

Penghapusan *stopwords* bertujuan untuk menghilangkan kata yang dianggap tidak memiliki makna atau tidak relevan yang terdapat pada dokumen. Contoh *stopwords* yang ada pada Bahasa Indonesia seperti “yang”, “dan”, “di”, “itu”, “adapun”, “agak” dan sebagainya. Kata-kata umum tersebut tidak mempunyai nilai pada sebuah dokumen sehingga kata-kata yang termasuk dalam kamus *stopwords* dihilangkan sehingga ukuran data juga akan berkurang.

2.7.6 Stemming

Stemming merupakan proses yang bertujuan untuk menemukan kata dasar dari sebuah kata dengan menghapus imbuhan (*affixes*), termasuk awalan (*prefixes*), sisipan (*infixes*), akhiran (*suffixes*), serta kombinasi dari awalan dan akhiran (*confixes*) pada kata turunan. Tujuan utama dari proses *stemming* ini adalah mengubah bentuk kata menjadi kata dasar sesuai dengan bahasa Indonesia yang baik dan benar. Menurut algoritma Nazief & Adriani memiliki tahap-tahap sebagai berikut:

1. Mencari kata yang akan dilakukan *stemming* pada kamus, apabila ditemukan maka diasumsikan bahwa kata tersebut merupakan kata akar (*root word*), jika terbukti maka proses diberhentikan pada tahap pertama.
2. Menghilangkan *inflection suffixes*, sebuah kata yang mengandung *inflection suffixes* yaitu apabila memiliki imbuhan “-lah”, “-ku”, “-kah”, “-mu”, atau “-nya”, apabila kata berupa *particles* atau dalam kata lain yang mengandung “-lah”, “-kah”, atau “-pun”, maka langkah untuk menghilangkan *inflection suffixes* diulangi agar dapat menghilangkan *possessive pronouns* yang termasuk imbuhan *possessive pronouns* adalah “-ku”, “-mu”, atau “-nya”.
3. Menghapus *derivation suffixes* atau imbuhan turunan yang termasuk pada kata imbuhan turunan adalah “-i”, “-an”, atau “-kan”.
4. Menghapus *derivational prefix* atau imbuhan yang berada pada awal kata yang dimaksud dengan *derivational prefix* adalah “be-”, “di-”, “ke-”, “me-”, “pe-”, “se-”, dan “te-”.
5. Apabila 4 langkah tersebut telah dilakukan tetapi belum berhasil menemukan kata dasar maka algoritma ini akan melakukan analisis apakah kata tersebut termasuk ke dalam tabel ambiguitas kolom terakhir.
6. Apabila belum berhasil maka algoritma akan dikembalikan pada kata aslinya.

Stemming pada *python* dilakukan melalui kelas *StemmerFactory* yaitu sebuah kelas yang terdapat pada *library* yang bernama “*Sastrawi*” dan kompatibel dengan *input* berbahasa Indonesia yang mana *library* ini akan lebih dulu di-*import* sebelum meng-*import* kelas *StemmerFactory*.

2.8 Lexicon Based

Metode berbasis *Lexicon* merupakan metode yang sederhana, layak, dan praktis untuk analisis sentimen dari data media sosial. Data yang cocok dengan metode *Lexicon Based* yaitu data kuesioner, data *Twitter*, data *Facebook*, atau media sosial lainnya yang berupa opini pelanggan tentang suatu produk atau pelayanan jasa (Matulatuwa et al., 2017).

Metode *Lexicon Based* didasarkan pada asumsi bahwa orientasi sentimen kontekstual adalah jumlah dari orientasi sentimen setiap kata atau frasa. Metode ini dapat digunakan untuk melakukan ekstraksi sentimen dari blog dengan mengombinasikan *lexical knowledge* dan klasifikasi teks. Metode *Lexicon* dapat dibuat secara manual atau diperluas secara otomatis dari *seed of words* (Matulatuwa et al., 2017).

Penentuan label sentimen dilakukan pada data teks berupa kalimat yang memiliki kata pada kamus *lexicon* yang terdiri dari kata negatif dan positif. Kata yang teridentifikasi dalam kamus *lexicon* akan dihitung skornya sesuai dengan jumlah kata pada setiap teks atau kalimat (Ismail et al., 2023).

$$S_{positive} = \sum_{i \in t}^n positive\ score_i \dots\dots\dots(2.5)$$

$$S_{negative} = \sum_{i \in t}^n negative\ score_i \dots\dots\dots(2.6)$$

$S_{positive}$ adalah bobot dari kalimat yang didapatkan melalui penjumlahan n skor polaritas kata opini positif dan $S_{negative}$ adalah bobot dari kalimat yang didapatkan melalui penjumlahan n skor polaritas kata opini negatif. Oleh karena itu, dari persamaan nilai sentimen dalam satu kalimat diperoleh persamaan untuk menentukan orientasi sentimen dengan perbandingan jumlah nilai positif, negatif, dan netral (Ismail et al., 2023).

$$Sentence_{sentiment} \begin{cases} positive\ if\ S_{positive} > S_{negative} \\ neutral\ if\ S_{positive} = S_{negative} \\ negative\ if\ S_{positive} < S_{negative} \end{cases} \dots\dots\dots(2.7)$$

Jika dalam suatu teks memiliki jumlah kata positif lebih banyak dari kata negatif, maka data teks tersebut akan dilabeli sentimen positif. Jika dalam suatu teks memiliki jumlah kata positif lebih sedikit dari kata negatif, maka data teks tersebut akan dilabeli sentimen negatif. Jika dalam suatu teks memiliki jumlah kata positif sama dengan kata negatif, maka data teks tersebut akan dilabeli sentimen netral (Ismail et al., 2023).

2.9 Grid Search Cross Validation

Grid Search adalah sebuah metode optimasi yang menjadikan titik-titik *grid* yang berjarak sama, lalu menghitung ukuran akurasi untuk setiap parameter tersebut sehingga ditemukan titik parameter yang paling optimal atau titik yang memiliki ukuran akurasi terbaik (Satriani et al., 2019). *Grid Search* dalam implementasinya dikombinasikan dengan metode *cross validation*. *Cross Validation* adalah sebuah metode pengembangan dari model *split validation* yang mana validasinya mengukur *training error* dengan *test data* (Muiz, n.d.). Penentuan nilai *cross validation* yang direkomendasikan untuk pemilihan model terbaik adalah 10 (Wibowo, 2017).

2.10 K-Nearest Neighbors (KNN)

KNN adalah algoritma *machine learning classifier* populer yang paling sederhana yang pertama kali diperkenalkan oleh T. Cover dan P. Hart pada tahun 1967 dimana algoritma ini mengklasifikasikan kelas sampel berdasarkan kelas tetangga terdekatnya (Fajri et al., 2020). *K-Nearest Neighbors* sendiri memiliki prinsip sederhana, bekerja berdasarkan jarak terpendek dari sampel uji ke sampel latih (Sari, 2020). Algoritma KNN bekerja dengan cara menghitung jarak tiap titik pada data uji dengan data latih seluruh kelas. Lalu, diurutkan dari jarak terdekat ke jarak terjauh dan akan dipilih jarak terdekat antara data uji dengan data latih sejumlah k . Kelas yang memiliki jarak terdekat dengan data uji akan menjadi kelas data uji tersebut (Raschka, 2018). Adapun tahapan proses yang dilakukan pada KNN menurut Abdillah (2023) adalah sebagai berikut:

1. Hitung jarak antara sampel yang tidak diketahui dengan semua sampel pada set data pelatihan menggunakan rumus jarak yang dipilih, didalam kasus ini digunakan *Cosine Similarity*.
2. Pilih k tetangga terdekat dari sampel yang tidak diketahui berdasarkan jarak yang telah dihitung.
3. Hitung label kelas mayoritas dari k tetangga terdekat. Dalam kasus klasifikasi biner, label mayoritas dapat dihitung dengan menghitung frekuensi masing-masing kelas pada k tetangga terdekat dan memilih kelas dengan frekuensi yang paling tinggi. Dalam kasus klasifikasi multikelas, label mayoritas dihitung dengan metode voting, yaitu dengan menghitung jumlah suara setiap kelas pada k tetangga terdekat dan memilih kelas dengan jumlah suara terbanyak.
4. Label kelas mayoritas menjadi kelas hasil klasifikasi untuk sampel yang tidak diketahui.

Berdasarkan tahapan proses tersebut, langkah yang paling penting adalah menghitung jarak atau tingkat kemiripan data dengan setiap data latih yang ada menggunakan rumus jarak *Cosine Similarity*. Sistem akan mengurutkan nilai jarak dari yang tertinggi sampai terendah. Kelebihan dari algoritma *Cosine Similarity* adalah tidak terpengaruh pada panjang pendeknya suatu dokumen dan memiliki tingkat akurasi yang tinggi. Tahapan pada *Cosine similarity* adalah sebagai berikut (Abdillah, 2023):

1. Kalikan bobot dari setiap *term* pada D1 dengan setiap *term* dari semua dokumen data latih yang ada.
2. Hasil perkalian D1 dengan setiap dokumen kemudian dijumlahkan.
3. Hitung hasil kuadrat dari masing-masing *term* dalam setiap dokumen (termasuk D1) kemudian jumlahkan lalu diakarkan.
4. Lakukan pembagian antara hasil dari langkah nomor 2 dengan langkah nomor 3. Maka, didapatkan nilai *Cosine Similarity*. Berikut adalah rumus *Cosine Similarity*:

$$\cos(\theta_{QD}) = \frac{\sum_{i=1}^n Q_i D_i}{\sqrt{\sum_{i=1}^n (Q_i)^2} \sqrt{\sum_{i=1}^n (D_i)^2}} \dots\dots\dots(2.8)$$

Keterangan:

$\cos(\theta_{QD})$: kemiripan Q terhadap dokumen D
Q	: data uji
D	: data latih
n	: jumlah data latih

2.11 Validasi dan Pengujian

Validasi dan pengujian sangat diperlukan untuk menilai kinerja dari sebuah sistem. Kinerja proses klasifikasi menggambarkan seberapa baik sistem dalam melakukan klasifikasi data. Pengukuran kinerja proses tersebut dapat dilakukan menggunakan *confusion matrix* (Abdillah, 2023).

Confusion matrix merupakan suatu matriks yang digunakan untuk menganalisa keakuratan dari model klasifikasi yang dibuat untuk mengidentifikasi data dengan kelas yang berbeda (Afrillia et al., 2022). Pengujian dengan *confusion matrix* ini digunakan untuk menghitung nilai *true positive*, *false positive*, *true negative*, serta *false negative* tergantung banyaknya kelas klasifikasi yang nantinya dapat digunakan untuk pengukuran nilai akurasi, presisi, serta *recall*. Melakukan pengukuran tingkat akurasi dapat mengetahui seberapa baik performa model klasifikasi tersebut. Bentuk dari *confusion matrix* adalah tabel dengan empat atau lebih kombinasi berdasarkan kelas-kelas pada penelitian ini yang berbeda antara nilai prediksi dan nilai aktual. Berikut merupakan tabel *confusion matrix*:

Tabel 2.1 Confusion Matrix

		<i>Predicted</i>		
		-1 (<i>Negative</i>)	+1 (<i>Positive</i>)	0 (<i>Neutral</i>)
<i>Actual</i>	-1 (<i>Negative</i>)	TN	FP	FL
	+1 (<i>Positive</i>)	FN	TP	FL2
	0 (<i>Neutral</i>)	FN2	FP2	TL

Keterangan:

TN (<i>True Negative</i>)	: Prediksi benar bernilai negatif
TP (<i>True Positive</i>)	: Prediksi benar bernilai positif
TL (<i>True Neutral</i>)	: Prediksi benar bernilai netral
FN (<i>False Negative</i>)	: Positif terprediksi negatif
FN2 (<i>False Negative 2</i>)	: Netral terprediksi negatif
FP (<i>False Positive</i>)	: Negatif terprediksi positif
FP2 (<i>False Positive 2</i>)	: Netral terprediksi positif
FL (<i>False Neutral</i>)	: Negatif terprediksi netral
FL2 (<i>False Neutral 2</i>)	: Positif terprediksi netral

Tabel *confusion matrix* di atas dapat digunakan dalam perhitungan *performance matrix* yang bertujuan untuk mengukur model yang digunakan untuk dapat memperoleh nilai *accuracy*, *recall*, *precision*, dan *f1-score*.

1. Accuracy

Akurasi merupakan nilai yang menunjukkan kedekatan antar nilai prediksi dan nilai aktual. Perhitungan akurasi dengan cara membagi jumlah data yang akan diklasifikasi secara tepat dengan total sampel data *testing* yang diuji. Berikut merupakan rumus perhitungan nilai akurasi:

$$Accuracy = \frac{TP+TN+TL}{TN+FP+FL+FN+TP+FL2+FN2+FP2+TL} \dots\dots\dots(2.9)$$

2. Recall

Recall merupakan perbandingan jumlah data yang dilakukan prediksi pada kelas yang benar dengan jumlah data yang diharapkan berada pada kelas yang benar. *Recall* dikatakan sebagai tingkat keberhasilan model dalam menemukan informasi. Berikut merupakan rumus perhitungan nilai *recall*:

$$Recall\ Positive = \frac{TP}{TP+FN+FL2} \dots\dots\dots(2.10)$$

$$Recall\ Negative = \frac{TN}{TN+FP+FL} \dots\dots\dots (2.11)$$

$$Recall\ Neutral = \frac{TL}{TL+FN2+FP2} \dots\dots\dots (2.12)$$

$$Recall = \frac{Recall\ Positive+Recall\ Negative+Recall\ Neutral}{3} \times 100\% \dots\dots\dots(2.13)$$

3. *Precision*

Precision merupakan tingkat akurasi antar informasi yang diinginkan pengguna serta respon sistem. Nilai presisi menunjukkan data positif yang diklasifikasi dengan tepat kemudian dilakukan pembagian dengan jumlah data positif yang diklasifikasi. Berikut merupakan rumus perhitungan nilai presisi:

$$Preci\ Positive = \frac{TP}{TP+FP+FP2} \dots\dots\dots(2.14)$$

$$Preci\ Negative = \frac{TN}{TN+FN+FN2} \dots\dots\dots(2.15)$$

$$Preci\ Neutral = \frac{TL}{TL+FL+FL2} \dots\dots\dots(2.16)$$

$$Precision = \frac{Preci\ Positive+Preci\ Negative+Preci\ Neutral}{3} \times 100\% \dots\dots\dots(2.17)$$

4. *F1-Score*

F1-Score merupakan perbandingan rata-rata *precision* dan *recall* yang telah dibobotkan. Nilai terbaik *F1-Score* adalah 1 dan nilai terburuknya yaitu 0. Perhitungan yang didapatkan berupa informasi bahwa model klasifikasi memiliki *precision* dan *recall* yang baik. Berikut merupakan rumus perhitungan nilai *F1-Score*:

$$F1\ Positive = 2 \times \frac{Preci\ Positive \times Recall\ Positive}{Preci\ Positive+Recall\ Positive} \dots\dots\dots(2.18)$$

$$F1\ Negative = 2 \times \frac{Preci\ Negative \times Recall\ Negative}{Preci\ Negative+Recall\ Negative} \dots\dots\dots(2.19)$$

$$F1\ Neutral = 2 \times \frac{Preci\ Neutral \times Recall\ Neutral}{Preci\ Neutral+Recall\ Neutral} \dots\dots\dots(2.20)$$

$$F1\ Score = \frac{F1\ Positive+F1\ Negative+F1\ Neutral}{3} \times 100\% \dots\dots\dots(2.21)$$

2.12 Studi Pustaka

Penelitian di bawah ini merupakan penelitian-penelitian yang telah dilakukan sebelumnya dan berkaitan dengan penelitian tugas akhir ini, sehingga menjadi referensi dalam penelitian ini.

Tabel 2.2 Studi Pustaka

No	Penulis	Judul	Metode	Hasil
1.	Elfansyah et al., 2024.	Perbandingan Metode K-Nearest Neighbor (KNN) Dan Naïve Bayes Terhadap Analisis Sentimen Pada Pengguna E-Wallet Aplikasi Dana Menggunakan Fitur Ekstraksi TF-IDF	Klasifikasi <i>KNN</i> dan <i>Naïve Bayes</i> . Pelabelan sentimen berbasis <i>Lexicon</i> dengan acuan kamus label oleh tenaga ahli bahasa (<i>expert</i>) dan <i>library Lexicon via Python</i> . Pembobotan kata menggunakan <i>TF-IDF</i> .	Data yang diberi label model <i>Lexicon</i> , akurasi <i>KNN</i> 78% dan <i>Naïve Bayes</i> 74%. Terkait nilai <i>k</i> , keputusan akhir sebagai parameter final yang diambil menurut penulis adalah <i>k</i> = 5 dikarenakan memberikan kinerja optimal tanpa penurunan akurasi yang signifikan.
2.	Alamsyah & Mulyati, 2023.	Implementasi Algoritme K-Nearest Neighbour Dan Lexicon Based Untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Gramedia Digital Pada Media Sosial Twitter	Klasifikasi <i>KNN</i> . Pelabelan sentimen berbasis <i>Lexicon</i> dengan acuan kamus label <i>InSet</i> . Pembobotan kata menggunakan <i>TF-IDF</i> .	<i>KNN</i> berhasil mencapai akurasi tertinggi sebesar 75.97% dengan nilai <i>k</i> = 3 dengan rasio pembagian data 60:40.
3.	Putri et al., 2023.	Analisis Sentimen dan Pemodelan Ulasan Aplikasi AdaKami Menggunakan Algoritma SVM dan KNN	Klasifikasi <i>KNN</i> dan <i>SVM</i> . Pelabelan sentimen berbasis <i>library Lexicon via Python</i> dan manual. Pembobotan kata menggunakan <i>TF-IDF</i> .	Rasio pembagian data 90:10, 80:20, 70:30, 60:40, 50:50. Model analisis sentimen dengan performa paling optimal menggunakan algoritma <i>SVM</i> dengan metode pelabelan manual dan proporsi pembagian data 90:10 dengan akurasi sebesar 93%, presisi 93%, <i>recall</i> 93%, dan <i>f1-score</i> 92%. Penelitian ini berkesimpulan bahwa <i>SVM</i> menghasilkan model dengan performa lebih optimal dibanding <i>KNN</i>

Tabel 2.3 Lanjutan Studi Pustaka

No	Penulis	Judul	Metode	Hasil
4.	Rahayu et al., 2022.	Implementasi Metode K-Nearest Neighbor (K-NN) untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Teknologi Finansial FLIP	Klasifikasi <i>KNN</i> . Pembobotan kata menggunakan <i>TF-IDF</i> . Pelabelan sentimen berbasis <i>Lexicon</i> dengan acuan kamus label <i>SentiWordnet</i> .	Rasio pembagian data 80:20 dengan algoritma klasifikasi <i>KNN</i> memperoleh akurasi sebesar 76.68%.
5.	Angel et al., 2024.	Analisis Sentimen dan Emosi dari Ulasan Google Maps untuk Layanan Rumah Sakit di Palangka Raya Menggunakan Machine Learning	Klasifikasi <i>KNN</i> , <i>Logistic Regression</i> , dan <i>Decision Tree</i> . Pelabelan sentimen berbasis <i>Lexicon Vader via Python</i> dan <i>NRC Lexicon</i> (emosi). Pembobotan kata menggunakan <i>TF-IDF</i> .	Akurasi tertinggi didapat oleh algoritma klasifikasi <i>Decision Tree</i> sebesar 92%, diikuti dengan <i>Logistic Regression</i> dengan akurasi 86%, dan <i>KNN</i> dengan akurasi 48%.
6.	Sholeha et al., 2022.	Analisis Sentimen Pada Agen Perjalanan Online Menggunakan Naïve Bayes dan K-Nearest Neighbor	Klasifikasi <i>KNN</i> dan <i>Naïve Bayes</i> . Pelabelan sentimen berbasis manual. Penghitungan frekuensi kata dengan <i>TF</i> .	Menurut penulis, berdasarkan 1500 data komentar dari 3 <i>fanpage Facebook</i> agen perjalanan online, ditemukan bahwa algoritma <i>KNN</i> memiliki akurasi yang lebih baik pada rata-rata dibandingkan <i>Naïve Bayes</i> dengan akurasi tertinggi 52.35%. Akurasi tertinggi kedua algoritma klasifikasi tersebut didapatkan saat seluruh data menggunakan huruf kecil.
7	Pramayasa et al., 2023.	Analisis Sentimen Program Mbkm Pada Media Sosial Twitter Menggunakan KNN Dan SMOTE	Klasifikasi <i>KNN</i> , penyeimbang kelas <i>SMOTE</i> .	Hasil performa metode <i>KNN</i> dengan nilai akurasi 68.81%, presisi 70.73%, <i>recall</i> 68.81%, serta <i>f1-score</i> 68.42%. Hasil optimasi menggunakan metode <i>SMOTE</i> mendapatkan hasil akurasi 76.13%, presisi 76.03%, <i>recall</i> 76.13%, serta <i>f1-score</i> 76.01%.

Tabel 2.4 Lanjutan Studi Pustaka

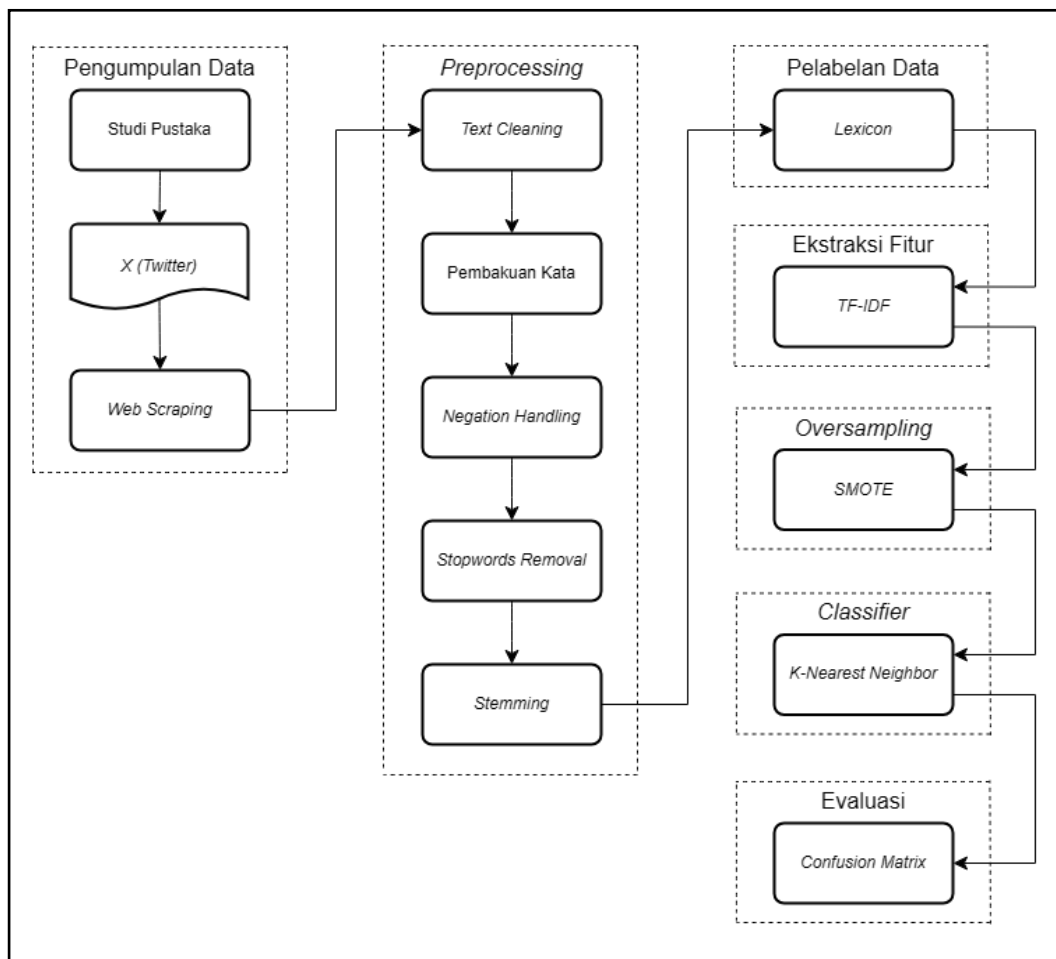
No	Penulis	Judul	Metode	Hasil
8.	Cahyana & Siregar, 2023.	Analisis Sentimen Pindah Ibu Kota Negara (IKN) Baru pada Twitter Menggunakan Algoritma Naive Bayes dan Support Vector Machine (SVM)	<i>SVM</i> dan <i>Naïve Bayes</i> , Pelabelan Manual Dosen Bahasa Indonesia, Pembobotan <i>TF-IDF</i> .	Hasil dari penelitian tersebut menunjukkan nilai akurasi sebesar 86.94%. Hal ini menunjukkan bahwa analisis sentimen pada penelitian ini memiliki nilai presisi rata-rata 96.24%, dan nilai recall 86.66%. Sedangkan penelitian menggunakan algoritme Support Vector Machine (SVM) menghasilkan nilai akurasi sebesar 90.81%. Hasil analisis sentimen pada penelitian ini memiliki nilai Nilai presisi rata-rata 90.12%, dan nilai recall 99.12%.
9.	Lestari et al., 2022.	Analisis Sentimen Masyarakat Indonesia terhadap Peminatan Ibu Kota Negara Indonesia pada Twitter	<i>SVM</i> , <i>Naïve Bayes</i> , dan <i>KNN</i> . Pelabelan Manual.	Algoritma SVM memperoleh nilai akurasi 85,71%, algoritma Naïve Bayes memperoleh nilai akurasi 76,70%, algoritma KNN memperoleh nilai akurasi 52,74%.
10.	Kusuma et al., 2023	Optimasi K-Nearest Neighbor dengan Grid Search CV pada Prediksi Kanker Paru-Paru	<i>KNN</i> , <i>Grid Search CV</i> , <i>SMOTE</i> .	Penerapan optimasi nilai <i>k</i> pada algoritma <i>KNN</i> menggunakan metode <i>grid search cv</i> berkesimpulan bahwa nilai <i>k</i> paling optimal yaitu 3 dengan perbandingan data latih 80% dan data uji 20%. Dengan nilai <i>k</i> = 3, algoritma <i>KNN</i> mampu memprediksi data dengan nilai akurasi 96%. Oleh karena itu, nilai <i>k</i> = 3 sangat baik digunakan untuk melatih model menggunakan objek dan <i>dataset</i> pada penelitian tersebut menggunakan algoritma <i>KNN</i> .

BAB III

METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM

3.1 Metodologi Penelitian

Bagian ini membahas mengenai metodologi penelitian serta pengembangan sistem yang akan dilakukan dalam penelitian ini yaitu menganalisis sentimen publik terhadap topik pemindahan ibu kota negara berdasarkan unggahan dalam bentuk teks di sosial media *X (Twitter)* dengan metode klasifikasi *KNN* dan pelabelan sentimen berbasis leksikon serta ekstraksi fitur *TF-IDF*. Metodologi penelitian merupakan sub bab yang menggambarkan alur kerja serta tahapan pada penelitian ini. Tahapan metodologi penelitian pada tugas akhir ini yaitu pengumpulan data, *preprocessing* data, pelabelan data, pembobotan kata, penyeimbangan kelas, pembuatan model, pengujian model, pengujian sistem, serta analisis dan visualisasi hasil. Berikut merupakan tahapan pada metodologi penelitian ini dapat dilihat pada gambar 3.1.



Gambar 3.1 Metodologi Penelitian

3.1.1 Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh data yang nantinya akan diolah menjadi informasi yang mudah dipahami dan dapat digunakan sebagai dasar pengambilan keputusan. Data yang terdapat dalam penelitian ini dikumpulkan dengan melakukan *web scraping* dari media sosial *X (Twitter)*.

a. Studi Pustaka

Studi pustaka merupakan langkah awal dalam proses penelitian. Kegiatan ini bertujuan untuk mencari dan memperoleh informasi terkait metode, topik, serta masalah yang menjadi fokus penelitian. Studi pustaka ini didasarkan pada referensi dari jurnal, makalah, prosiding, buku, dan sumber terpercaya lainnya. Studi pustaka dilakukan untuk memperkuat argumentasi dalam penelitian yang sedang dilakukan.

b. *X (Twitter)*

Data yang digunakan dalam penelitian ini diambil dari media sosial *X*. *X* atau yang sebelumnya dikenal dengan sebutan *Twitter* merupakan media sosial yang memungkinkan pengguna untuk berbagi pesan singkat yang disebut *tweet* atau cuitan. Media sosial ini digunakan secara luas oleh berbagai kalangan, mulai dari individu hingga organisasi besar untuk menyampaikan pendapat, berita, dan informasi lainnya. Platform ini memberikan peluang untuk dilakukannya analisis sentimen publik terhadap pemindahan ibu kota negara dan mengamati reaksi pengguna terhadap berbagai topik, termasuk ulasan tentang layanan tertentu.

c. *Web Scraping*

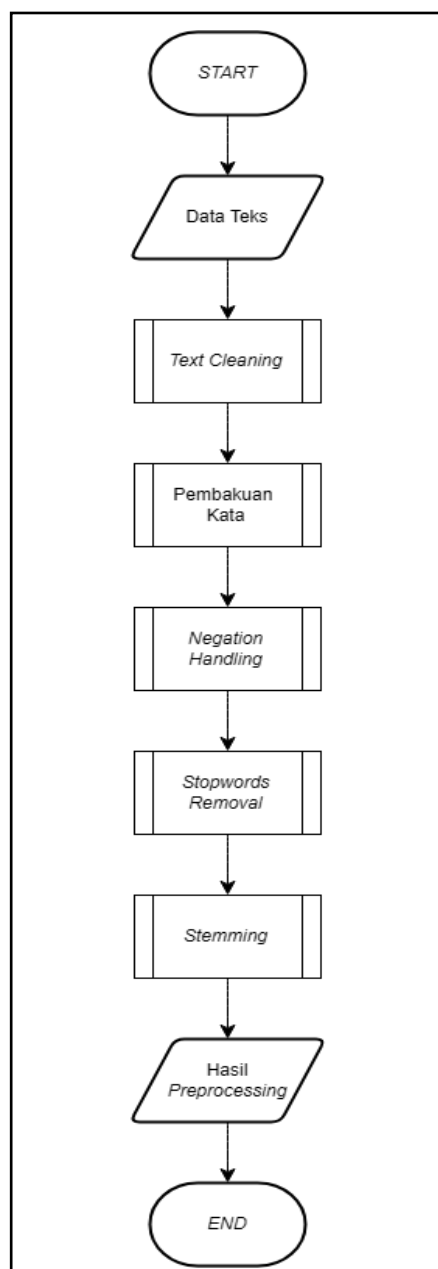
Pengumpulan data dilakukan dengan cara *web scraping* terhadap media sosial *X (Twitter)*. Data yang berhasil dikumpulkan akan dilakukan *preprocessing*. Data yang digunakan pada penelitian ini merupakan data berbahasa Indonesia dengan total data berjumlah 5341 baris *tweet* menggunakan 9 kata kunci yang tertera pada tabel 3.1.

Tabel 3.1 Kata Kunci

No.	Kata Kunci	Jumlah Data
1	<i>ikn</i>	1005
2	<i>ibu kota baru</i>	223
3	<i>ibu kota nusantara</i>	15
4	<i>ibu kota pindah</i>	1005
5	<i>pemindahan ibu kota</i>	1010
6	<i>ibukota baru</i>	212
7	<i>ibukota nusantara</i>	804
8	<i>ibukota pindah</i>	554
9	<i>pemindahan ibukota</i>	513

3.1.2 Preprocessing

Tahap penting dalam pengolahan data yang bertujuan untuk membersihkan dan mempersiapkan data agar siap digunakan dalam analisis sentimen. Pada tahapan ini, berbagai langkah dilakukan, seperti menghapus data yang tidak relevan (misalnya tanda baca, angka, dan simbol khusus), mengubah teks menjadi format standar (seperti konversi huruf besar menjadi huruf kecil), menghilangkan kata-kata umum yang tidak memiliki makna khusus (*stopwords*), serta melakukan *stemming* untuk mengembalikan kata ke bentuk dasarnya. *Preprocessing* sangat penting untuk meningkatkan akurasi model dalam memahami dan menganalisis data secara efektif, terutama dalam tugas-tugas yang melibatkan analisis teks seperti analisis sentimen. Berikut merupakan *flowchart* dari *preprocessing* dapat dilihat pada gambar 3.2 serta data sampel dari tahapan pengumpulan data pada tabel 3.2.



Gambar 3.2 *Flowchart Preprocessing*

Tabel 3.2 Contoh Hasil Pengumpulan Data

ID	Dokumen
D1	@meraaahputiiih Pindah ibu kota, semangat baru buat Indonesia yang semakin maju
D2	Jokowi minta daerah sekitar IKN jadi tulang punggung ketahanan pangan buat ibu kota baru. Ini jadi peluang besar buat daerah seperti Kalimantan Timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi. ðŸ™ https://t.co/2tHAdGwaiz
D3	@MerMerlyea gara2 siapa ini yg memaksa bikin ibu kota baru ðŸ™ƒðŸ™ƒ kalo negara ni ga mampu knpa di paksa sii ðŸ™¶â€ðŸ™€ü, □
D4	@TirtoID Ditawarin sosok yg cuma bikin taman aja ada analisanya,sampe mikirin mau dibikin park / garden,malah milih ngelanjutin mindah ibu kota dulu baru pusing kemudian.

a. *Text Cleaning*

Proses ini bertujuan untuk membersihkan data dengan menghapus elemen-elemen yang tidak relevan, seperti tanda baca, angka, emoji, *url*, atau karakter khusus lainnya yang tidak memiliki makna signifikan untuk analisis teks. *Cleansing* sangat membantu untuk fokus pada elemen-elemen yang relevan pada data untuk analisis lebih lanjut.

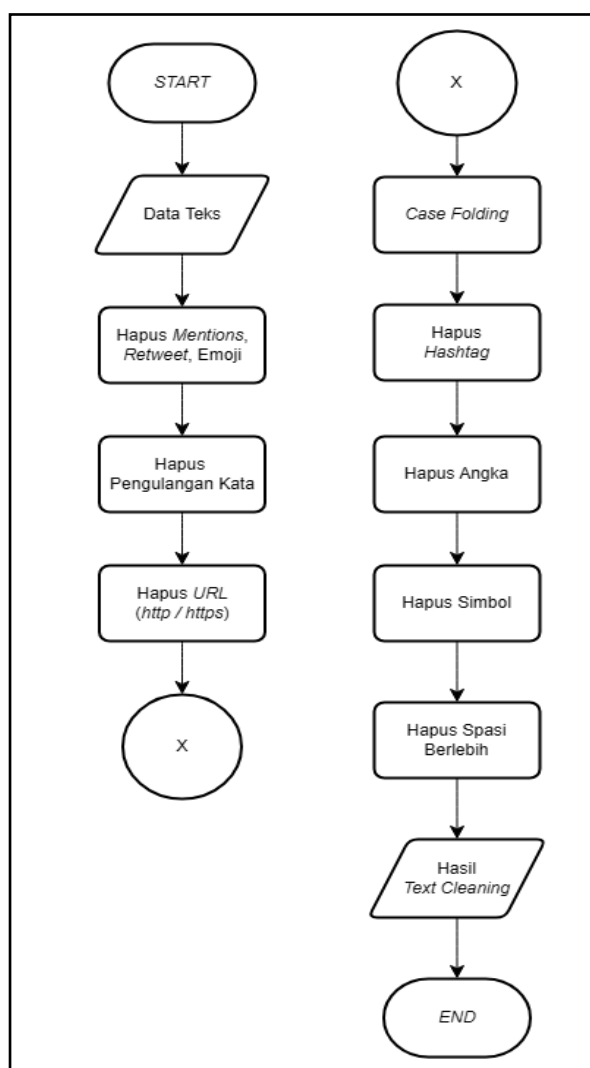
Tahap ini juga mengubah seluruh teks menjadi huruf kecil (*lower case*). Hal tersebut dilakukan untuk menghilangkan perbedaan antara huruf besar dan huruf kecil, karena dalam analisis berbasis teks, kata dengan huruf besar dan kecil dianggap berbeda.

Tabel 3.3 Data Sebelum *Text Cleaning*

ID	Dokumen
D1	@meraaahputiiih Pindah ibu kota, semangat baru buat Indonesia yang semakin maju
D2	Jokowi minta daerah sekitar IKN jadi tulang punggung ketahanan pangan buat ibu kota baru. Ini jadi peluang besar buat daerah seperti Kalimantan Timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi. ðŸ™ https://t.co/2tHAdGwaiz
D3	@MerMerlyea gara2 siapa ini yg memaksa bikin ibu kota baru ðŸ™ƒðŸ™ƒ kalo negara ni ga mampu knpa di paksa sii ðŸ™¶â€ðŸ™€ü, □
D4	@TirtoID Ditawarin sosok yg cuma bikin taman aja ada analisanya,sampe mikirin mau dibikin park / garden,malah milih ngelanjutin mindah ibu kota dulu baru pusing kemudian.

Tabel 3.4 Data Hasil *Text Cleaning*

ID	Dokumen
D1	pindah ibu kota semangat baru buat indonesia yang semakin maju
D2	jokowi minta daerah sekitar ibukota negara baru jadi tulang punggung ketahanan pangan buat ibu kota baru ini jadi peluang besar buat daerah seperti kalimantan timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi
D3	gara siapa ini yg memaksa bikin ibu kota baru kalo negara ni ga mampu knpa di paksa sii
D4	ditawarin sosok yg cuma bikin taman aja ada analisanya sampe mikirin mau dibikin park garden malah milih ngelanjutin mindah ibu kota dulu baru pusing kemudian



Gambar 3.3 *Flowchart Text Cleaning*

b. Pembakuan Kata

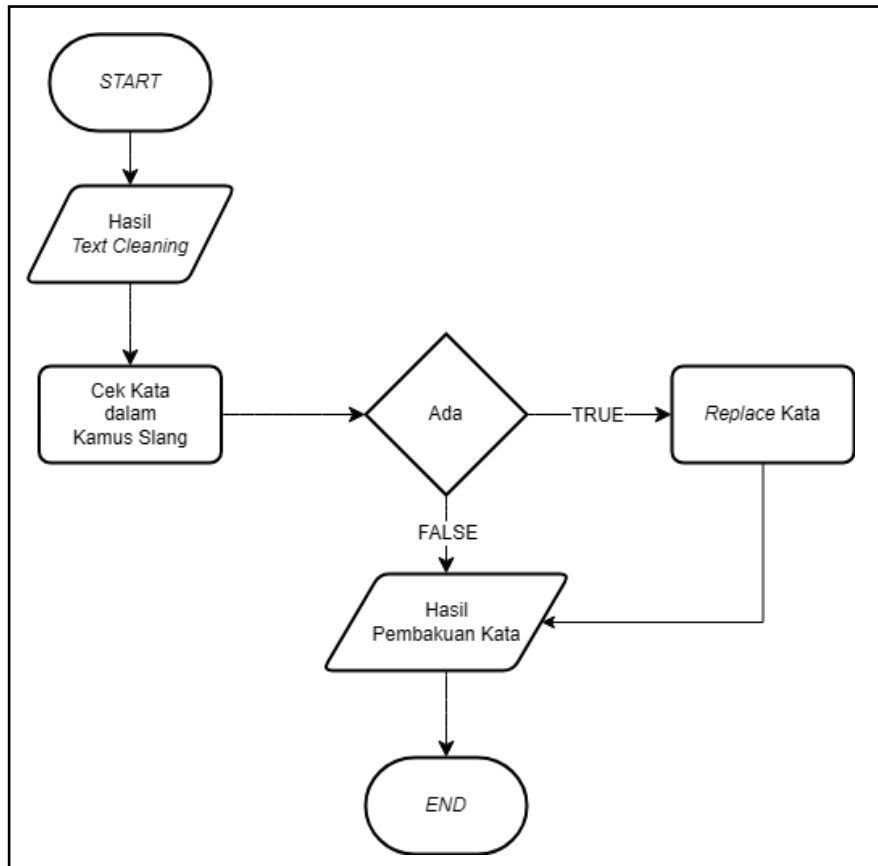
Tahapan ini mengganti kata slang atau gaul menjadi kata baku dengan menggunakan kamus kata slang yang didapatkan dari beberapa sumber di internet agar lebih mudah terbaca oleh kamus *lexicon* saat dilakukan pelabelan sentimen.

Tabel 3.5 Data Sebelum Pembakuan Kata

ID	Dokumen
D1	pindah ibu kota semangat baru buat indonesia yang semakin maju
D2	jokowi minta daerah sekitar ibukota negara baru jadi tulang punggung ketahanan pangan buat ibu kota baru ini jadi peluang besar buat daerah seperti kalimantan timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi
D3	gara siapa ini yg memaksa bikin ibu kota baru kalo negara ni ga mampu knpa di paksa sii
D4	ditawarin sosok yg cuma bikin taman aja ada analisisnya sampe mikirin mau dibikin park garden malah milih ngelanjutin mindah ibu kota dulu baru pusing kemudian

Tabel 3.6 Data Hasil Pembakuan Kata

ID	Dokumen
D1	pindah ibu kota semangat baru buat indonesia yang semakin maju
D2	jokowi meminta daerah sekitar ibukota negara baru jadi tulang punggung ketahanan pangan buat ibu kota baru ini jadi peluang besar buat daerah seperti kalimantan timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi
D3	gara siapa ini yang memaksa bikin ibu kota baru kalau negara ini tidak mampu kenapa di paksa sih
D4	ditawarin sosok yang cuma bikin taman saja ada analisisnya sampai memikirkan mau dibikin park garden bahkan memilih ngelanjutin mindah ibu kota dulu baru pusing kemudian



Gambar 3.4 Flowchart Pembakuan Kata

c. *Negation Handling*

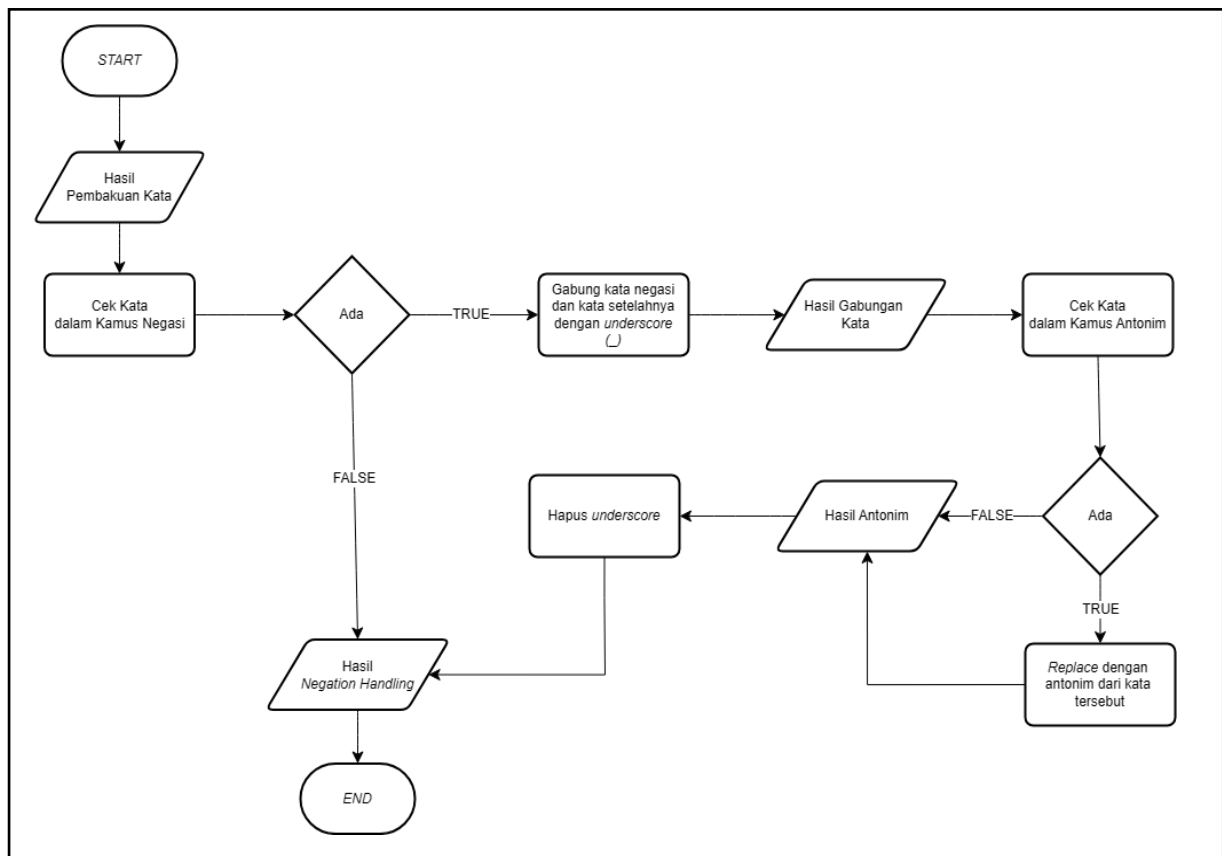
Tahapan *negation handling* mengubah bentuk kata yang memiliki makna yang sama tanpa kata negasi seperti “tidak” dan “belum”. Contohnya seperti “tidak bagus” yang maknanya sama dengan “jelek”, atau “tidak susah” yang sama maknanya dengan “mudah”.

Tabel 3.7 Data Sebelum *Negation Handling*

ID	Dokumen
D1	pindah ibu kota semangat baru buat indonesia yang semakin maju
D2	jokowi meminta daerah sekitar ibukota negara baru jadi tulang punggung ketahanan pangan buat ibu kota baru ini jadi peluang besar buat daerah seperti kalimantan timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi
D3	gara siapa ini yang memaksa bikin ibu kota baru kalau negara ini tidak mampu kenapa di paksa sih
D4	ditawarin sosok yang cuma bikin taman saja ada analisisnya sampai memikirkan mau dibikin park garden bahkan memilih ngelanjutin mindah ibu kota dulu baru pusing kemudian

Tabel 3.8 Data Hasil *Negation Handling*

ID	Dokumen
D1	pindah ibu kota semangat baru buat indonesia yang semakin maju
D2	jokowi meminta daerah sekitar ibukota negara baru jadi tulang punggung ketahanan pangan buat ibu kota baru ini jadi peluang besar buat daerah seperti kalimantan timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi
D3	gara siapa ini yang memaksa bikin ibu kota baru kalau negara ini tidak mampu kenapa di paksa sih
D4	ditawarin sosok yang cuma bikin taman saja ada analisisnya sampai memikirkan mau dibikin park garden bahkan memilih ngelanjutin mindah ibu kota dulu baru pusing kemudian



Gambar 3.5 Flowchart *Negation Handling*

d. *Stopwords Removal*

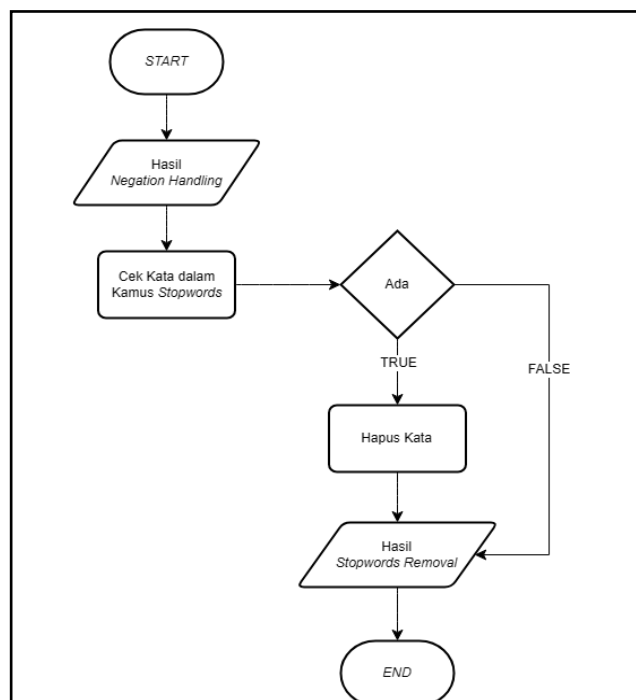
Pada tahap ini, kata-kata umum yang sering muncul tetapi tidak memiliki makna khusus dalam konteks analisis, seperti “dan”, “atau”, “dengan” akan dihilangkan. Kata-kata ini disebut *stopwords* dan biasanya tidak memberikan kontribusi signifikan dalam penentuan makna teks.

Tabel 3.9 Data Sebelum *Stopwords Removal*

ID	Dokumen
D1	pindah ibu kota semangat baru buat indonesia yang semakin maju
D2	jokowi meminta daerah sekitar ibukota negara baru jadi tulang punggung ketahanan pangan buat ibu kota baru ini jadi peluang besar buat daerah seperti kalimantan timur buat kembangkan sektor pertanian dan perikanan secara lebih terintegrasi
D3	gara siapa ini yang memaksa bikin ibu kota baru kalau negara ini tidak mampu kenapa di paksa sih
D4	ditawarin sosok yang cuma bikin taman saja ada analisisnya sampai memikirkan mau dibikin park garden bahkan memilih ngelanjutin mindah ibu kota dulu baru pusing kemudian

Tabel 3.10 Data Hasil *Stopwords Removal*

ID	Dokumen
D1	pindah kota semangat indonesia maju
D2	jokowi daerah ibukota negara tulang punggung ketahanan pangan kota peluang daerah kalimantan timur kembangkan sektor pertanian perikanan terintegrasi
D3	gara memaksa bikin kota negara paksa sih
D4	ditawarin sosok bikin taman analisisnya memikirkan dibikin park garden memilih ngelanjutin mindah kota pusing



Gambar 3.6 Flowchart *Stopwords Removal*

e. *Stemming*

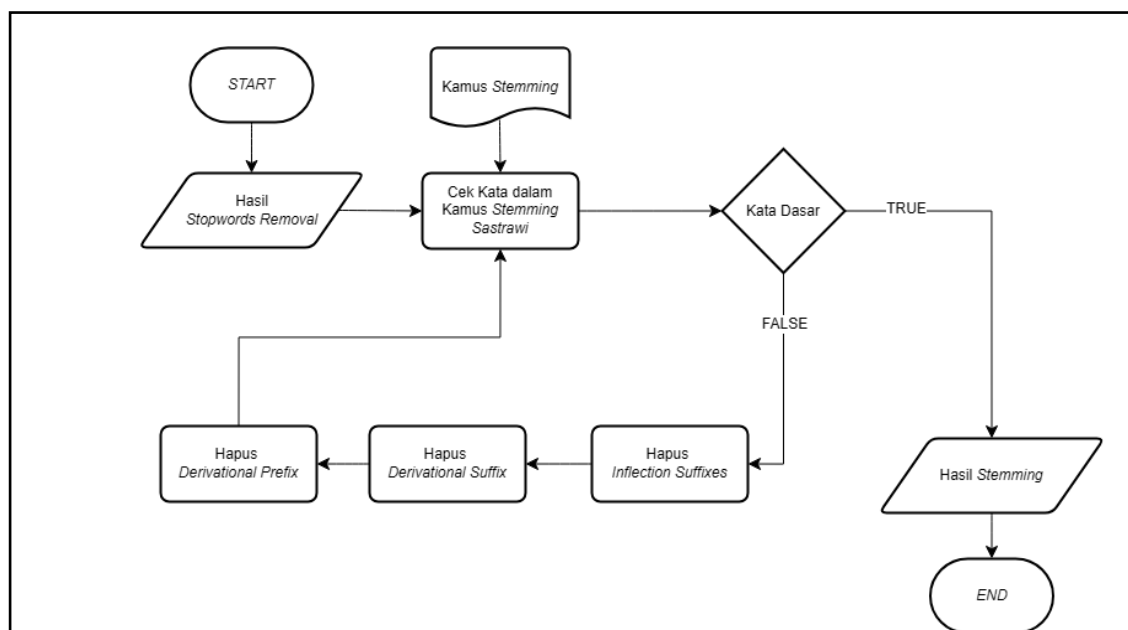
Proses *stemming* merupakan proses mengubah kata-kata dalam teks menjadi bentuk dasarnya atau kata dasar, dengan membuang awalan, akhiran, atau sisipan. Seperti *inflection suffixes* (-lah, -kah, -tah, -pun, -ku, dsb), *derivational suffix* (-an, -kan, -i), dan *derivational prefix* (-be, -di, -ke, -me, -pe, -se, -te). Proses *stemming* yang ada dalam penelitian ini menggunakan algoritma Nazief & Adriani untuk teks berbahasa Indonesia.

Tabel 3.11 Data Sebelum *Stemming*

ID	Dokumen
D1	pindah kota semangat indonesia maju
D2	jokowi daerah ibukota negara tulang punggung ketahanan pangan kota peluang daerah kalimantan timur kembangkan sektor pertanian perikanan terintegrasi
D3	gara memaksa bikin kota negara paksa sih
D4	ditawarin sosok bikin taman analisisnya memikirkan dibikin park garden memilih ngelanjutin mindah kota pusing

Tabel 3.12 Data Hasil *Stemming*

ID	Dokumen
D1	pindah kota semangat indonesia maju
D2	jokowi daerah ibukota negara tulang punggung tahan pangan kota peluang daerah kalimantan timur kembang sektor tani ikan integrasi
D3	gara paksa bikin kota negara paksa sih
D4	ditawarin sosok bikin taman analisisnya pikir bikin park garden pilih ngelanjutin mindah kota pusing



Gambar 3.7 Flowchart Stemming

3.1.3 Sentiment Labelling

Pada tahapan ini, digunakan kamus atau *lexicon InSet* atau *Indonesia Sentiment Lexicon* dan *library* yang terdapat pada *python* yaitu *vader lexicon*. *InSet* memiliki 2 kamus daftar kata yang masing-masing merupakan daftar kata kamus positif dan negatif dengan nilai atau bobot berbentuk bilangan bulat untuk setiap kata yang tersedia. Setiap kata pada kalimat atau teks yang terdapat dalam *dataset* dicocokkan dengan daftar kata pada masing-masing *lexicon*, jika kata pada kalimat tersebut ditemukan dalam *lexicon*, maka setiap kata yang cocok akan dihitung nilai sentimen atau *compound score* berdasarkan hasil perhitungan probabilitas sentimen pada teks tersebut. Hasil akhirnya dihitung dengan menjumlahkan skor sentimen dari kamus negatif dan positif. Kategori sentimen kemudian dilakukan pengondisian berdasarkan penjumlahan skor sentimen dari kedua kamus tersebut, jika skor total lebih besar dari nol, maka teks diberi label sentimen positif. Selain itu, jika skor total kurang dari nol, maka teks diberi label sentimen negatif. Apabila skor adalah nol, maka label sentimen dianggap netral.

Tabel 3.13 Data Sebelum Pelabelan Sentimen Leksikon

ID	Dokumen	Sentimen
D1	pindah kota semangat indonesia maju	?

Pada kamus positif *InSet* didapatkan bobot dari beberapa kata pada data sampel, kata “pindah” bernilai 1, “semangat” bernilai 4, “maju” bernilai 3. Kemudian pada kamus negatifnya, kata “pindah” bernilai -3, “kota” bernilai -1, dan “semangat” bernilai -1. Jika kata tidak ada di kedua kamus, dianggap netral dan tidak berpengaruh ke skor. Setelah itu dilakukan normalisasi skor positif dan negatif agar menjadi nilai probabilitas dengan membagi masing-masing skor positif dan negatif dengan total bobot absolut semua kata yang ditemukan. Setelah didapatkan nilai probabilitas, dilakukan perhitungan *compound score* dengan menggunakan rumus *sigmoid* yang menggabungkan skor positif dan negatif. Hasil dari pengurangan skor positif dan skor negatif dibagi dengan akar kuadrat dari total *token* yang mana terdapat kata “pindah”, “semangat”, dan “maju” dengan total 3 *token* yang kemudian hasilnya dioperasikan dengan *tanh()* untuk memetakan nilainya ke dalam rentang (-1, 1) untuk mendapatkan *compound score* dari kamus positif. Terakhir, hasil *compound score* dari kedua kamus tersebut dijumlahkan dan hasilnya sebagai *input* untuk menentukan apakah sentimen dari kalimat tersebut termasuk ke positif, negatif, atau netral.

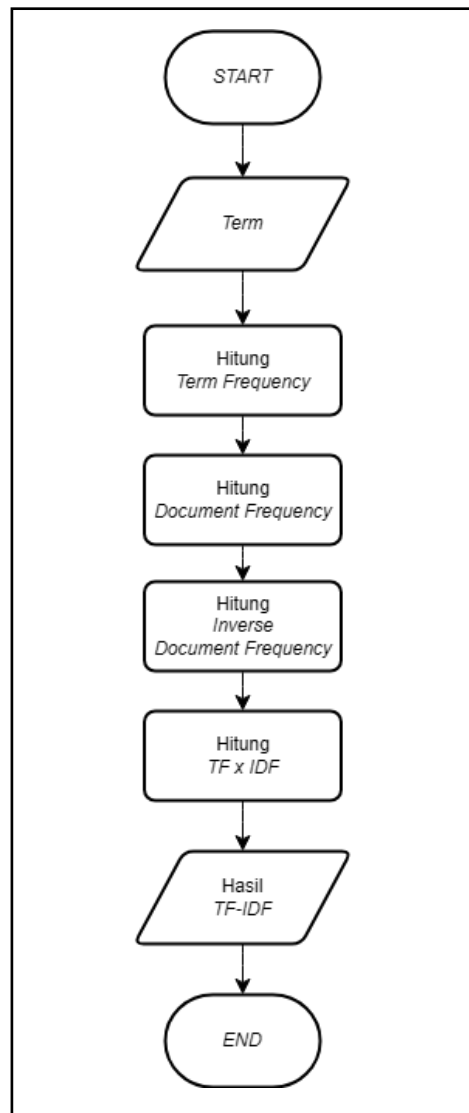
Tabel 3.14 Data Hasil Pelabelan Sentimen Leksikon

ID	Dokumen	Sentimen
D1	pindah kota semangat indonesia maju	Positif

3.1.4 Pembobotan *TF-IDF*

Pembobotan kata dengan menggunakan metode *Term Frequency-Inverse Document Frequency (TF-IDF)* adalah teknik yang digunakan untuk mengekstraksi fitur dalam analisis berbasis teks, termasuk analisis sentimen. *TF-IDF* adalah cara untuk mengukur seberapa penting sebuah kata dalam sebuah dokumen relatif terhadap keseluruhan kumpulan dokumen atau korpus. Metode ini memberikan bobot lebih tinggi pada kata-kata yang sering muncul dalam suatu dokumen, tetapi jarang ditemukan dalam dokumen lain di korpus, karena dianggap lebih signifikan.

Setiap kata dalam dokumen akan diberi bobot menggunakan nilai *TF-IDF*. Kata-kata dengan bobot *TF-IDF* tinggi dianggap lebih relevan dan memiliki peran lebih besar dalam mendeskripsikan dokumen tersebut. Pembobotan ini membantu mengurangi dampak dari kata-kata umum yang sering muncul tetapi tidak memberikan banyak informasi (*stopwords*), sementara memberi fokus lebih pada kata-kata unik yang lebih menggambarkan isi dari dokumen.



Gambar 3.8 *Flowchart TF-IDF*

Gambar 3.8 menunjukkan tahapan perhitungan *TF-IDF*, yang dimulai dengan memilih *term* yang akan dihitung. Selanjutnya, dilakukan penghitungan frekuensi kemunculan kata dalam dokumen (*TF*), diikuti dengan menghitung jumlah dokumen yang mengandung kata tersebut (*DF*). Setelah itu, nilai *IDF* dihitung, lalu tahap terakhir adalah menghitung nilai *TF-IDF* dengan mengalikan hasil *TF* dan *IDF*.

1. Menghitung *Term Frequency (TF)*

Term Frequency (TF) adalah frekuensi kemunculan suatu istilah dalam sebuah dokumen. Semakin sering istilah tersebut muncul dalam dokumen, semakin tinggi bobotnya. Berikut merupakan contoh data yang akan digunakan untuk perhitungan manual *TF-IDF* dapat dilihat pada tabel 3.15.

Tabel 3.15 Contoh Dokumen Perhitungan *Term Frequency*

ID	Dokumen	Sentimen
D1	pindah kota semangat indonesia maju	Netral
D2	jokowi daerah ibukota negara tulang punggung tahan pangan kota peluang daerah kalimantan timur kembang sektor tani ikan integrasi	Positif
D3	gara paksa bikin kota negara paksa sih	Negatif
D4	ditawarin sosok bikin taman analisisnya pikir bikin park garden pilih ngelanjutin mindah kota pusing	Negatif

Selanjutnya yaitu menghitung nilai *Term Frequency (TF)* dari masing-masing dokumen yang telah dilakukan *preprocessing* pada setiap *term* yang ada. Proses perhitungan *TF* dengan memberikan nilai 1 apabila *term* tersebut terdapat pada komentar dan sebaliknya. Proses dilakukan pada dokumen 1 (*D1*) hingga dokumen 4 (*D4*). Hasil *TF* dapat dilihat pada tabel 3.16 dan tabel 3.17.

Tabel 3.16 Hasil *Term Frequency*

No.	Term	TF			
		D1	D2	D3	D4
1	analisisnya	0	0	0	1
2	bikin	0	0	1	1
3	daerah	0	1	0	0
4	ditawarin	0	0	0	1
5	gara	0	0	1	0
6	garden	0	0	0	1
7	ibukota	0	1	0	0
8	ikan	0	1	0	0
9	indonesia	1	0	0	0
10	integrasi	0	1	0	0
11	jokowi	0	1	0	0
12	kalimantan	0	1	0	0
13	kembang	0	1	0	0
14	kota	1	1	1	1

Tabel 3.17 Lanjutan Hasil *Term Frequency*

No.	Term	TF			
		D1	D2	D3	D4
15	maju	1	0	0	0
16	mindah	0	0	0	1
17	negara	0	1	1	0
18	ngelanjutin	0	0	0	1
19	paksa	0	0	1	0
20	pangan	0	1	0	0
21	park	0	0	0	1
22	peluang	0	1	0	0
23	pikir	0	0	0	1
24	pilih	0	0	0	1
25	pindah	1	0	0	0
26	punggung	0	1	0	0
27	pusing	0	0	0	1
28	sektor	0	1	0	0
29	semangat	1	0	0	0
30	sih	0	0	1	0
31	sosok	0	0	0	1
32	tahan	0	1	0	0
33	taman	0	0	0	1
34	tani	0	1	0	0
35	timur	0	1	0	0
36	tulang	0	1	0	0

Dapat dilihat pada tabel 3.16 dan tabel 3.17 merupakan *term* yang telah diproses perhitungan *TF*, dengan diberi nilai 1 apabila *term* tersebut terdapat pada dokumen, dan nilai 0 apabila *term* tersebut tidak ada dalam dokumen.

2. Menghitung Nilai *Inverse Document Frequency (IDF)*

Setelah perhitungan *Term Frequency (TF)* selesai, langkah berikutnya adalah menghitung *Inverse Document Frequency (IDF)*, yaitu menghitung seberapa banyak *term* muncul di seluruh dokumen. Rumus *IDF* dihitung menggunakan persamaan:

$$idf_t = \log \frac{N}{df_t} \dots\dots\dots(3.1)$$

Berikut merupakan hasil perhitungan *IDF* dapat dilihat pada tabel 3.18 dan tabel 3.19.

Tabel 3.18 Hasil *IDF*

No.	Term	TF				DF	N/df(t)	IDF
		D1	D2	D3	D4			
1	analisnya	0	0	0	1	1	4.0	1.916291
2	bikin	0	0	1	1	2	2.0	1.510826
3	daerah	0	1	0	0	1	4.0	1.916291

Tabel 3.19 Lanjutan Hasil *IDF*

<i>No.</i>	<i>Term</i>	<i>TF</i>				<i>DF</i>	<i>N/df(t)</i>	<i>IDF</i>
		<i>D1</i>	<i>D2</i>	<i>D3</i>	<i>D4</i>			
4	ditawarin	0	0	0	1	1	4.0	1.916291
5	gara	0	0	1	0	1	4.0	1.916291
6	garden	0	0	0	1	1	4.0	1.916291
7	ibukota	0	1	0	0	1	4.0	1.916291
8	ikan	0	1	0	0	1	4.0	1.916291
9	indonesia	1	0	0	0	1	4.0	1.916291
10	integrasi	0	1	0	0	1	4.0	1.916291
11	jokowi	0	1	0	0	1	4.0	1.916291
12	kalimantan	0	1	0	0	1	4.0	1.916291
13	kembang	0	1	0	0	1	4.0	1.916291
14	kota	1	1	1	1	4	1.0	1.000000
15	maju	1	0	0	0	1	4.0	1.916291
16	mindah	0	0	0	1	1	4.0	1.916291
17	negara	0	1	1	0	2	2.0	1.510826
18	ngelanjutin	0	0	0	1	1	4.0	1.916291
19	paksa	0	0	1	0	1	4.0	1.916291
20	pangan	0	1	0	0	1	4.0	1.916291
21	park	0	0	0	1	1	4.0	1.916291
22	peluang	0	1	0	0	1	4.0	1.916291
23	pikir	0	0	0	1	1	4.0	1.916291
24	pilih	0	0	0	1	1	4.0	1.916291
25	pindah	1	0	0	0	1	4.0	1.916291
26	punggung	0	1	0	0	1	4.0	1.916291
27	pusing	0	0	0	1	1	4.0	1.916291
28	sektor	0	1	0	0	1	4.0	1.916291
29	semangat	1	0	0	0	1	4.0	1.916291
30	sih	0	0	1	0	1	4.0	1.916291
31	sosok	0	0	0	1	1	4.0	1.916291
32	tahan	0	1	0	0	1	4.0	1.916291
33	taman	0	0	0	1	1	4.0	1.916291
34	tani	0	1	0	0	1	4.0	1.916291
35	timur	0	1	0	0	1	4.0	1.916291
36	tulang	0	1	0	0	1	4.0	1.916291

Tabel 3.18 dan tabel 3.19 merupakan hasil perhitungan nilai *IDF* atau seberapa sering suatu *term* muncul pada dokumen-dokumen tersebut.

3. Menghitung Nilai *TF-IDF*

Setelah didapatkan nilai *TF*, *DF*, dan *IDF* maka langkah selanjutnya yaitu menghitung nilai *TF-IDF* melalui persamaan berikut:

$$W_{at} = tf_{at} \times idf_t \dots\dots\dots(3.2)$$

W merupakan hasil dari perhitungan TF dikalikan dengan IDF ($TF \times IDF$), maka hasil perhitungan $TF-IDF$ dari dokumen diatas dapat dilihat pada tabel 3.20.

Tabel 3.20 Hasil $TF-IDF$

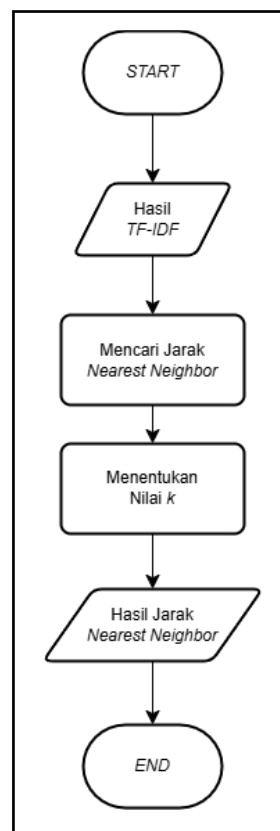
No.	Term	W_{dt}			
		$D1$	$D2$	$D3$	$D4$
1.	analisnya	0	0	0	0.269594
2.	bikin	0	0	0.287590	0.425103
3.	daerah	0	0.460117	0	0
4.	ditawarin	0	0	0	0.269594
5.	gara	0	0	0.364771	0
6.	garden	0	0	0	0.269594
7.	ibukota	0	0.230058	0	0
8.	ikan	0	0.230058	0	0
9.	indonesia	0.483802	0	0	0
10.	integrasi	0	0.230058	0	0
11.	jokowi	0	0.230058	0	0
12.	kalimantan	0	0.230058	0	0
13.	kembang	0	0.230058	0	0
14.	kota	0.252468	0.120054	0.190352	0.140685
15.	maju	0.483802	0	0	0
16.	mindah	0	0	0	0.269594
17.	negara	0	0.181381	0.287590	0
18.	ngelanjutin	0	0	0	0.269594
19.	paksa	0	0	0.729543	0
20.	pangan	0	0.230058	0	0
21.	park	0	0	0	0.269594
22.	peluang	0	0.230058	0	0
23.	pikir	0	0	0	0.269594
24.	pilih	0	0	0	0.269594
25.	pindah	0.483802	0	0	0
26.	punggung	0	0.230058	0	0
27.	pusing	0	0	0	0.269594
28.	sektor	0	0.230058	0	0
29.	semangat	0.483802	0	0	0
30.	sih	0	0	0.364771	0
31.	sosok	0	0	0	0.269594
32.	tahan	0	0.230058	0	0
33.	taman	0	0	0	0.269594
34.	tani	0	0.230058	0	0
35.	timur	0	0.230058	0	0
36.	tulang	0	0.230058	0	0

3.1.5 Penerapan Teknik *SMOTE*

Tahapan *Synthetic Minority Oversampling Technique* atau *SMOTE* adalah teknik yang digunakan untuk mengatasi ketidakseimbangan kelas dalam *dataset*. Teknik ini fokus pada kelas minoritas dalam *dataset*, dimana jumlah sampelnya jauh lebih sedikit dibandingkan dengan kelas mayoritas. Proses *SMOTE* melibatkan dua tahap yaitu mencari jarak *Nearest Neighbor*, dan menghasilkan data sintesis.

1. Mencari Jarak *Nearest Neighbor*

Pada setiap data dalam kelas minoritas, tetangga (*neighbor*) terdekatnya diidentifikasi dengan metode *KNN (K-Nearest Neighbor)* dengan nilai *k* yang ditentukan dan dengan nilai bobot kelas minoritas yang dihasilkan pada tahap *TF-IDF* sebelumnya. Untuk mencari nilai tetangga terdekat digunakan matriks *Euclidean Distance*. *Flowchart* untuk mencari jarak *nearest neighbor* dapat dilihat pada gambar 3.9 berikut:



Gambar 3.9 Flowchart Jarak *Nearest Neighbor* *SMOTE*

Contoh perhitungan mencari *nearest neighbor* pada *D2* ke *D3* pada *term* ‘paksa’ adalah sebagai berikut:

$$X_{knn} = \sqrt{(0 - 0.729543)^2} \dots\dots\dots(3.3)$$

$$X_{knn} = \sqrt{0.532232} \dots\dots\dots(3.4)$$

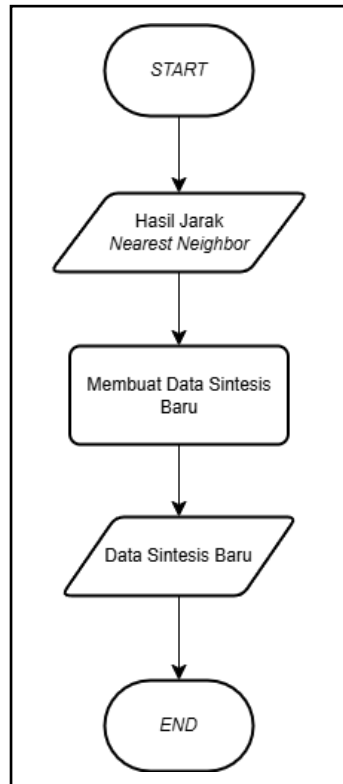
Proses perhitungan jarak *nearest neighbor* pada semua *term* pada kelas minor positif dengan contoh $k = 2$. Hasil jarak *nearest neighbor* dapat dilihat pada tabel 3.21 berikut:

Tabel 3.21 Hasil Jarak Nearest Neighbor SMOTE

No.	Term	Jarak Kelas Minor Positif					
		D2 ke data		D3 ke data		D4 ke data	
		D3	D4	D2	D4	D2	D3
1	analisnya	0	0.073	0	0.073	0.073	0.073
2	bikin	0.083	0.181	0.083	0.019	0.181	0.019
3	daerah	0.212	0.212	0.212	0	0.212	0
4	ditawarin	0	0.073	0	0.073	0.073	0.073
5	gara	0.133	0	0.133	0.133	0	0.133
6	garden	0	0.073	0	0.073	0.073	0.073
7	ibukota	0.053	0.053	0.053	0	0.053	0
8	ikan	0.053	0.053	0.053	0	0.053	0
9	indonesia	0	0	0	0	0	0
10	integrasi	0.053	0.053	0.053	0	0.053	0
11	jokowi	0.053	0.053	0.053	0	0.053	0
12	kalimantan	0.053	0.053	0.053	0	0.053	0
13	kembang	0.053	0.053	0.053	0	0.053	0
14	kota	0.005	0.0004	0.005	0.002	0.0004	0.002
15	maju	0	0	0	0	0	0
16	mindah	0	0.073	0	0.073	0.073	0.073
17	negara	0.011	0.033	0.011	0.083	0.033	0.083
18	ngelanjutin	0	0.073	0	0.073	0.073	0.073
19	paksa	0.532	0	0.532	0.532	0	0.532
20	pangan	0.053	0.053	0.053	0	0.053	0
21	park	0	0.073	0	0.073	0.073	0.073
22	peluang	0.053	0.053	0.053	0	0.053	0
23	pikir	0	0.073	0	0.073	0.073	0.073
24	pilih	0	0.073	0	0.073	0.073	0.073
25	pindah	0	0	0	0	0	0
26	punggung	0.053	0.053	0.053	0	0.053	0
27	pusing	0	0.073	0	0.073	0.073	0.073
28	sektor	0.053	0.053	0.053	0	0.053	0
29	semangat	0	0	0	0	0	0
30	sih	0.133	0	0.133	0.133	0	0.133
31	sosok	0	0.073	0	0.073	0.073	0.073
32	tahan	0.053	0.053	0.053	0	0.053	0
33	taman	0	0.073	0	0.073	0.073	0.073
34	tani	0.053	0.053	0.053	0	0.053	0
35	timur	0.053	0.053	0.053	0	0.053	0
36	tulang	0.053	0.053	0.053	0	0.053	0
Jumlah		1.851	1.971	1.851	1.705	1.971	1.705
Akar		1.361	1.404	1.361	1.306	1.404	1.306
$k = 2$		NN1	NN2	NN2	NN1	NN2	NN1

2. Pembentukan Data Sintesis

Pada tahap pembentukan data sintesis, dilakukan dengan cara memilih nilai *Nearest Neighbor* secara acak yang paling dekat dari setiap data. Data sintesis dibuat sepanjang garis vektor yang menghubungkan data asli dengan tetangga terdekatnya. Jumlah data sintesis yang dihasilkan bergantung pada nilai N yang telah ditentukan atau perbedaan antara jumlah data kelas minor dan mayor. *Flowchart* untuk membuat data sintesis baru dapat dilihat pada gambar 3.10 berikut:



Gambar 3.10 Flowchart Membuat Data Sintesis Baru SMOTE

Dalam mencari nilai data sintesis baru dilakukan persamaan:

$$X_{syn} = x_i + (x_{knn} - x_i) \times \delta \dots\dots\dots(3.5)$$

Dengan δ adalah nilai acak antara 0 sampai 1, digunakan nilainya 0,5. Hasil perhitungan data sintesis baru dapat dilihat pada tabel 3.22 berikut:

Tabel 3.22 Hasil Data Sintesis *SMOTE*

No.	Term	Data Sintesis Kelas Minor Positif		
		<i>D2syn</i>	<i>D3syn</i>	<i>D4syn</i>
1	analisnya	0.0365	0.0365	0.171297
2	bikin	0.0655	0.153295	0.222052
3	daerah	0.336058	0	0
4	ditawarin	0.0365	0.0365	0.171297
5	gara	0.0665	0.248886	0.0665
6	garden	0.0365	0.0365	0.171297
7	ibukota	0.141529	0.0265	0
8	ikan	0.141529	0.0265	0
9	indonesia	0	0	0
10	integrasi	0.141529	0.0265	0
11	jokowi	0.141529	0.0265	0
12	kalimantan	0.141529	0.0265	0
13	kembang	0.141529	0.0265	0
14	kota	0.060227	0.096176	0.071343
15	maju	0	0	0
16	mindah	0.0365	0.0365	0.171297
17	negara	0.10819	0.160295	0.0165
18	ngelanjutin	0.0365	0.0365	0.171297
19	paksa	0.266	0.630772	0.266
20	pangan	0.141529	0.0265	0
21	park	0.0365	0.0365	0.171297
22	peluang	0.141529	0.0265	0
23	pikir	0.0365	0.0365	0.171297
24	pilih	0.0365	0.0365	0.171297
25	pindah	0	0	0
26	punggung	0.141529	0.0265	0
27	pusing	0.0365	0.0365	0.171297
28	sektor	0.141529	0.0265	0
29	semangat	0	0	0
30	sih	0.0665	0.248886	0.0665
31	sosok	0.0365	0.0365	0.171297
32	tahan	0.141529	0.0265	0
33	taman	0.0365	0.0365	0.171297
34	tani	0.141529	0.0265	0
35	timur	0.141529	0.0265	0
36	tulang	0.141529	0.0265	0

Sehingga hasil dokumen dari data sintesis untuk kelas minor positif dapat dilihat pada tabel 3.23 berikut:

Tabel 3.23 Dokumen Data Sintesis *SMOTE*

Data Sintesis Minor Positif		
<i>D2syn</i>	<i>D3syn</i>	<i>D4syn</i>
analisnya	analisnya	analisnya
bikin	bikin	bikin
daerah	ditawarin	ditawarin
ditawarin	gara	gara
gara	garden	garden
garden	ibukota	kota
ibukota	ikan	mindah
ikan	integrasi	negara
integrasi	jokowi	ngelanjutin
jokowi	kalimantan	paksa
kalimantan	kembang	park
kembang	kota	pikir
kota	mindah	pilih
mindah	negara	pusing
negara	ngelanjutin	sih
ngelanjutin	paksa	sosok
paksa	pangan	taman
pangan	park	
park	peluang	
peluang	pikir	
pikir	pilih	
pilih	punggung	
punggung	pusing	
pusing	sektor	
sektor	sih	
sih	sosok	
sosok	tahan	
tahan	taman	
taman	tani	
tani	timur	
timur	tulang	
tulang		

Setelah semua data sintesis diketahui hasil dan nilainya, maka berikut adalah perubahan tabel *TF-IDF* yang terdapat data sintesis:

Tabel 3.24 Hasil Perubahan *TF-IDF* Setelah *SMOTE*

No.	Term	TF-IDF						
		D1	D2	D3	D4	D5	D6	D7
1	analisnya	0	0	0	0.27	0.036	0.036	0.171
2	bikin	0	0	0.288	0.425	0.066	0.153	0.222
3	daerah	0	0.46	0	0	0.336	0	0
4	ditawarin	0	0	0	0.27	0.036	0.036	0.171
5	gara	0	0	0.365	0	0.067	0.249	0.067
6	garden	0	0	0	0.27	0.036	0.036	0.171
7	ibukota	0	0.23	0	0	0.142	0.027	0
8	ikan	0	0.23	0	0	0.142	0.027	0
9	indonesia	0.484	0	0	0	0	0	0
10	integrasi	0	0.23	0	0	0.142	0.027	0
11	jokowi	0	0.23	0	0	0.142	0.027	0
12	kalimantan	0	0.23	0	0	0.142	0.027	0
13	kembang	0	0.23	0	0	0.142	0.027	0
14	kota	0.252	0.12	0.19	0.141	0.06	0.096	0.071
15	maju	0.484	0	0	0	0	0	0
16	mindah	0	0	0	0.27	0.036	0.036	0.171
17	negara	0	0.183	0.288	0	0.108	0.16	0.017
18	ngelanjutin	0	0	0	0.27	0.036	0.036	0.171
19	paksa	0	0	0.73	0	0.266	0.631	0.266
20	pangan	0	0.23	0	0	0.142	0.027	0
21	park	0	0	0	0.27	0.036	0.036	0.171
22	peluang	0	0.23	0	0	0.142	0.027	0
23	pikir	0	0	0	0.27	0.036	0.036	0.171
24	pilih	0	0	0	0.27	0.036	0.036	0.171
25	pindah	0.484	0	0	0	0	0	0
26	punggung	0	0.23	0	0	0.142	0.027	0
27	pusing	0	0	0	0.27	0.036	0.036	0.171
28	sektor	0	0.23	0	0	0.142	0.027	0
29	semangat	0.484	0	0	0	0	0	0
30	sih	0	0	0.365	0	0.067	0.249	0.067
31	sosok	0	0	0	0.27	0.036	0.036	0.171
32	tahan	0	0.23	0	0	0.142	0.027	0
33	taman	0	0	0	0.27	0.036	0.036	0.171
34	tani	0	0.23	0	0	0.142	0.027	0
35	timur	0	0.23	0	0	0.142	0.027	0
36	tulang	0	0.23	0	0	0.142	0.027	0

3.1.6 Analisis Sentimen dengan Model *KNN*

Tahapan ini menentukan hasil analisis sentimen berdasarkan data yang telah melalui proses sebelumnya untuk mendapatkan hasil analisis sentimen menggunakan metode *K-Nearest Neighbor*. Berikut merupakan *flowchart* dari proses analisis sentimen dengan *KNN*.



Gambar 3.11 *Flowchart* Klasifikasi Model *KNN*

Analisis sentimen menggunakan model *KNN* berawal dari *input* data yang sudah melalui tahapan *resampling* dengan *SMOTE* dan sebelumnya dilakukan transformasi kata dengan *TF-IDF* untuk membobotkan data teks, sehingga setiap kata diwakili dalam bentuk vektor. Selanjutnya, dilakukan penghitungan jarak antara setiap sampel data uji terhadap seluruh data latih menggunakan *Cosine Similarity* untuk mengukur kemiripan. Berdasarkan hasil dari jarak yang sudah dihitung, data uji akan dikelompokkan dengan tetangga terdekat sejumlah *k*. Lalu, dari tetangga terdekat yang telah terpilih, kelas mayoritas akan ditetapkan sebagai prediksi kelas untuk data uji.

Tabel 3.25 Sampel Data *Input KNN*

ID	Dokumen	Fitur <i>TF-IDF</i>	Sentimen
<i>D1</i>	pindah kota semangat indonesia maju	[0.4, 0.2, 0.5, 0.5, 0.5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	Positif
<i>D2</i>	jokowi daerah ibukota negara tulang punggung tahan pangan kota peluang daerah kalimantan timur kembang sektor tani ikan integrasi	[0.2, 0.4, 0.2, 0.1, 0.2, 0.2, 0.2, 0.2, 0.1, 0.2, 0.4, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2]	Positif
<i>D3</i>	gara paksa bikin kota negara paksa sih	[0.3, 0.7, 0.3, 0.2, 0.3, 0.7, 0.3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	Negatif
<i>D4</i>	ditawarin sosok bikin taman analisisnya pikir bikin park garden pilih ngelanjutin mindah kota pusing	[0.2, 0.2, 0.4, 0.2, 0.2, 0.2, 0.4, 0.2, 0.2, 0.2, 0.2, 0.2, 0.1, 0.2, 0, 0, 0, 0]	Negatif
<i>D5</i>	pindah kota putus buruk	[0.4, 0.3, 0.5, 0.5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	?

Tabel di atas merupakan sampel data yang dipersiapkan sebelum proses klasifikasi data uji terhadap data latih dilakukan. *D1* sampai *D4* merupakan sampel data latih yang telah memiliki label atau kategori sentimen dan *D5* adalah data uji yang akan diklasifikasikan. Sampel data tersebut tidak melalui proses *SMOTE* dikarenakan telah dianggap seimbang antar kelasnya. Langkah pertama yang dilakukan adalah menentukan nilai k tetangga terdekat yaitu $k = 3$. Selanjutnya mencari nilai jarak dari data uji ke setiap data latih dengan *cosine similarity*, hasilnya dapat dilihat pada tabel di bawah ini.

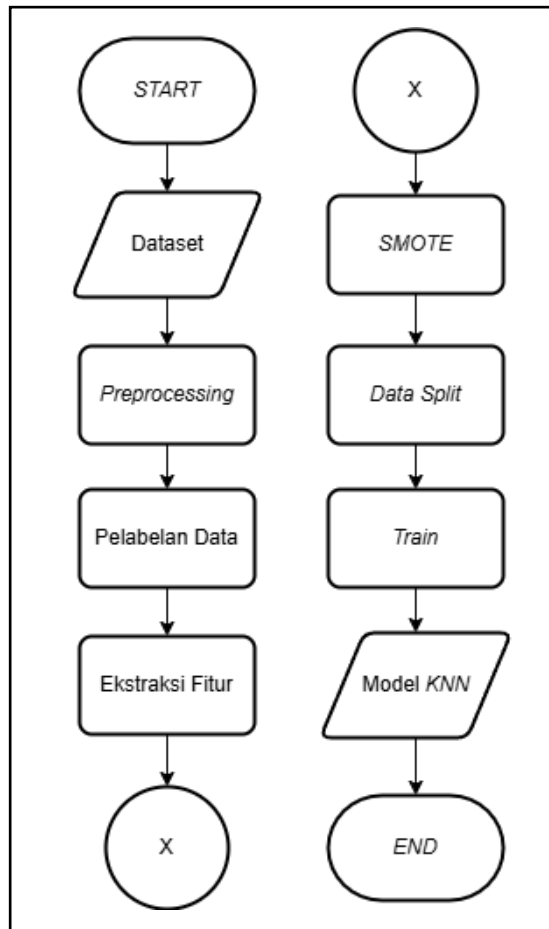
Tabel 3.26 Nilai Jarak Data Uji Ke Setiap Data Latih

ID	Dokumen	<i>Cosine Similarity</i>	Sentimen
<i>D1</i>	pindah kota semangat indonesia maju	0.831	Positif
<i>D2</i>	jokowi daerah ibukota negara tulang punggung tahan pangan kota peluang daerah kalimantan timur kembang sektor tani ikan integrasi	0.255	Positif
<i>D3</i>	gara paksa bikin kota negara paksa sih	0.490	Negatif
<i>D4</i>	ditawarin sosok bikin taman analisisnya pikir bikin park garden pilih ngelanjutin mindah kota pusing	0.623	Negatif

Mengacu pada nilai k yang telah ditentukan sebelumnya, diambil 3 nilai *cosine similarity* tertinggi yaitu *D1* dengan 0.831, *D4* dengan 0.623, dan *D3* dengan 0.490. Kelas data uji ditentukan dari kelas mayoritas yang terdapat pada 3 nilai *cosine similarity* tertinggi, pada contoh kasus ini hasil penentuan kelas data uji (*D5*) adalah Negatif.

3.1.7 Pembuatan Model Sentimen

Sebagai cara untuk mendapatkan hasil akhir dari penelitian ini, yaitu klasifikasi sentimen publik terhadap pemindahan ibu kota negara yang terdapat pada media sosial *X (Twitter)*, langkah yang diambil adalah melakukan klasifikasi menggunakan metode *K-Nearest Neighbor (KNN)*. Berikut merupakan *flowchart* pembuatan model sentimen dapat dilihat pada gambar 3.12.



Gambar 3.12 Flowchart Pembuatan Model Sentimen

Flowchart proses klasifikasi sentimen menggunakan metode *K-Nearest Neighbor* dimulai dengan mengambil hasil dari proses pengumpulan data, tahapan selanjutnya mencakup *preprocessing*, pelabelan, pembobotan kata, serta *resampling* yang digunakan untuk mempersiapkan data teks yang akan diklasifikasikan. Setelah itu, dilakukan pembagian data latih dan data uji masing-masing 80% dan 20% dari jumlah keseluruhan data. Kemudian, proses selanjutnya adalah pelatihan model dengan *input* dari hasil proses sebelumnya serta penentuan parameter terbaik sebagai variabel penting dalam pembuatan model sentimen dengan *K-Nearest Neighbor*. Hasil akhirnya adalah model analisis sentimen *KNN* untuk pengelompokan data opini publik mengenai pemindahan ibu kota negara ke dalam kategori sentimen yang tepat, yakni positif, netral, atau negatif. Pengulangan proses dari awal kembali bisa terjadi jika ada data baru yang dihasilkan dari proses pengumpulan data.

3.1.8 Pengujian

Pengujian sistem pada penelitian ini dilakukan menggunakan *confusion matrix* untuk mengevaluasi performa model yang diterapkan. Pengujian ini bertujuan untuk mengukur tingkat kinerja metode klasifikasi *K-Nearest Neighbor* yang digunakan dalam penelitian ini. Beberapa nilai kinerja yang diperoleh dari *confusion matrix* meliputi akurasi, presisi, dan *recall* model. Pada penelitian ini, data dibagi menjadi 80% data latih (*train*) dan 20% data uji (*test*). Pengujian dilakukan dengan menggunakan *library scikit-learn* yang terdapat pada *environment* bahasa pemrograman *python*, yang mana *library* tersebut dapat menghasilkan *confusion matrix* beserta nilai akurasi, presisi, dan *recall* secara otomatis.

3.2 Metode Pengembangan Sistem

Metode yang digunakan dalam pengembangan sistem ini adalah metode *Waterfall*. Metode ini merupakan metode yang terdiri dari tahap *requirements analysis*, *system and software design*, *implementation*, dan *system testing*. Berikut merupakan uraian terkait dengan masing-masing proses dari metode *Waterfall* yang digunakan sebagai metode pengembangan sistem:

3.2.1 Requirements Analysis

Tahapan awal ini merupakan tahapan yang berfokus pada pencarian kebutuhan apa saja yang bisa digunakan dalam proses pembuatan sistem ini diawali dengan mengidentifikasi kebutuhan keseluruhan yang akan diterapkan serta diimplementasikan menjadi sebuah perangkat lunak atau *software*. Tahap awal ini melibatkan proses pengumpulan data. Opini pengguna *X* yang terdapat pada kolom komentar media sosial *X* terkait dengan *keyword* yang digunakan dikumpulkan menggunakan teknik *scraping*. Kumpulan data opini yang diperoleh kemudian disimpan ke dalam suatu file untuk dianalisis lebih lanjut. Selain itu, pada tahapan ini juga dilakukan pengumpulan referensi studi literatur yang relevan dengan penelitian yang dilakukan.

1. Kebutuhan Fungsional

- a. Kemampuan sistem untuk melakukan *preprocessing* terhadap *input* yang diberikan.
- b. Sistem berhasil melakukan pelabelan data dengan *lexicon*.
- c. Sistem berhasil melakukan vektorisasi teks dengan *TF-IDF*.
- d. Sistem berhasil melakukan klasifikasi data terhadap kelas-kelas analisis sentimen dengan algoritma *K-Nearest Neighbor*.
- e. Sistem mampu untuk memberikan *output* nilai akurasi dari operasi perhitungan dengan seluruh *input* yang didapatkan.

2. Kebutuhan Non Fungsional

Implementasi sistem yang akan dibangun sesuai dengan rancangan yang telah dibuat memerlukan beberapa perangkat keras (*hardware*), perangkat lunak (*software*), dan pengguna sebagai dukungan dalam pengembangan sistem analisis. Berikut adalah analisis kebutuhan non-fungsional yang diperlukan untuk merancang sistem:

a. Kebutuhan *Hardware*

Berikut merupakan spesifikasi *hardware* yang akan digunakan dalam pembuatan sistem pada penelitian ini:

Tabel 3.27 Spesifikasi *Hardware*

No.	Hardware	Keterangan
1.	<i>Processor</i>	AMD Ryzen 5 5600H
2.	<i>RAM</i>	16 GB
3.	<i>SSD</i>	512 GB
4.	Perangkat Input dan Output	<i>Mouse, Keyboard, Trackpad</i>
5.	Koneksi	<i>WiFi, Seluler, LAN</i>

b. Kebutuhan *Software*

Berikut merupakan daftar *software* yang dipakai saat berjalannya proses penelitian:

Tabel 3.28 *Software*

No.	Software	Keterangan
1.	<i>Operating System</i>	Windows 11 64 bit, Ubuntu Linux
2.	<i>IDE</i>	Visual Studio Code, Google Colab, Kaggle Notebook
3.	Peramban	Google Chrome
4.	<i>Programming Language</i>	Python 3.12.4, Python 3.11.7
5.	Grafis Diagram	draw.io

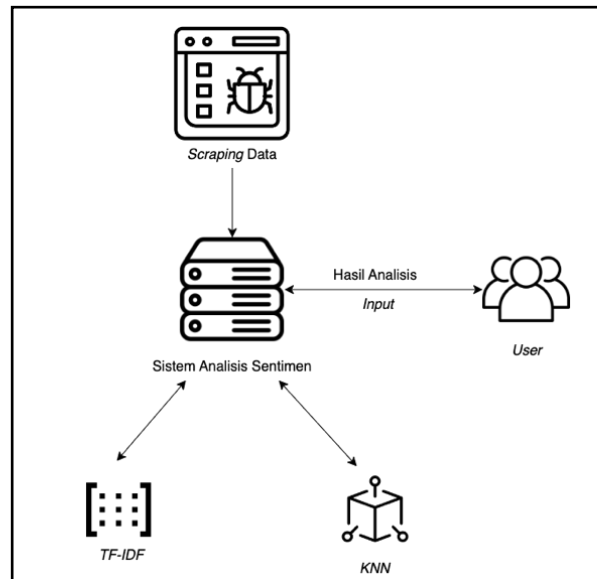
3.2.2 *System and Software Design*

Merupakan tahapan untuk melakukan perancangan mengenai sistem dan juga perangkat lunak yang akan dibangun dan digunakan dalam penelitian ini. Penting dan krusial merupakan alasan utama mengapa diperlukannya tahap ini. Adapun tahapannya sebagai berikut:

1. Perancangan Arsitektur

Beberapa arsitektur sistem yang akan dikembangkan dalam penelitian ini mencakup elemen-elemen seperti *user*, *web scraper*, model *KNN*, pelabelan sentimen berbasis *lexicon*, serta sistem analisis sentimen terhadap pemindahan ibu kota negara Indonesia. *User* dalam sistem ini adalah individu yang dapat menggunakan atau mengoperasikan sistem secara umum serta melakukan pengecekan terhadap *input* untuk mengetahui hasil analisis sentimennya.

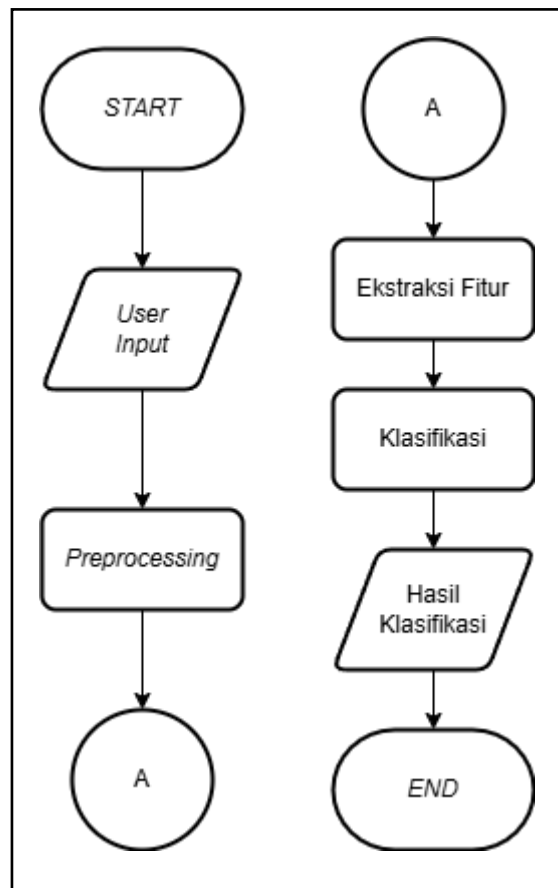
Pada sistem ini, terdapat beberapa tahap dimana *dataset* komentar publik terhadap pemindahan ibu kota negara diperoleh melalui proses *scraping* dari media sosial *X (Twitter)*. Data tersebut kemudian melalui tahapan *preprocessing* dan pelabelan sentimen menggunakan kamus *lexicon*. Setelah tahapan sebelumnya selesai, sistem melanjutkan dengan proses penganalisisan sentimen dengan model *KNN* yang telah melalui proses pelatihan (*training*), validasi (*validation*), dan pengujian (*testing*). Ketika model *KNN* sudah siap digunakan, sistem akan menampilkan hasil analisis sentimen untuk *user*. Ilustrasi arsitektur sistem dapat dilihat pada Gambar 3.13 berikut.



Gambar 3.13 Arsitektur Sistem

2. Perancangan Proses

Perancangan ini menggambarkan tahapan dalam analisis sentimen pada data teks menggunakan algoritma klasifikasi *K-Nearest Neighbor* serta pembobotan kata dengan *TF-IDF*. Tahapan diawali dengan menentukan *input* berupa teks yang berkaitan dengan opini mengenai pemindahan ibu kota negara. Setelah itu, dilakukan *preprocessing* terhadap data berfungsi sebagai penghilangan elemen-elemen yang tidak diperlukan atau berpotensi menjadi variabel yang membingungkan model. Kemudian, data yang telah melalui *preprocessing* diberi bobot menggunakan metode *TF-IDF*, yang menonjolkan kata-kata yang relevan dengan memberikan bobot yang lebih tinggi pada kata yang sering muncul di dokumen tertentu tetapi jarang di seluruh korpus. Hasil dari proses sebelumnya kemudian digunakan sebagai *input* bagi model *KNN* untuk mengklasifikasikan sentimen dalam data tersebut. Tahapan akhir adalah klasifikasi sentimen, dimana model *KNN* yang telah dilatih mengelompokkan data ke dalam sentimen positif, netral, atau negatif. Setelah klasifikasi selesai, seluruh prosesnya berakhir.



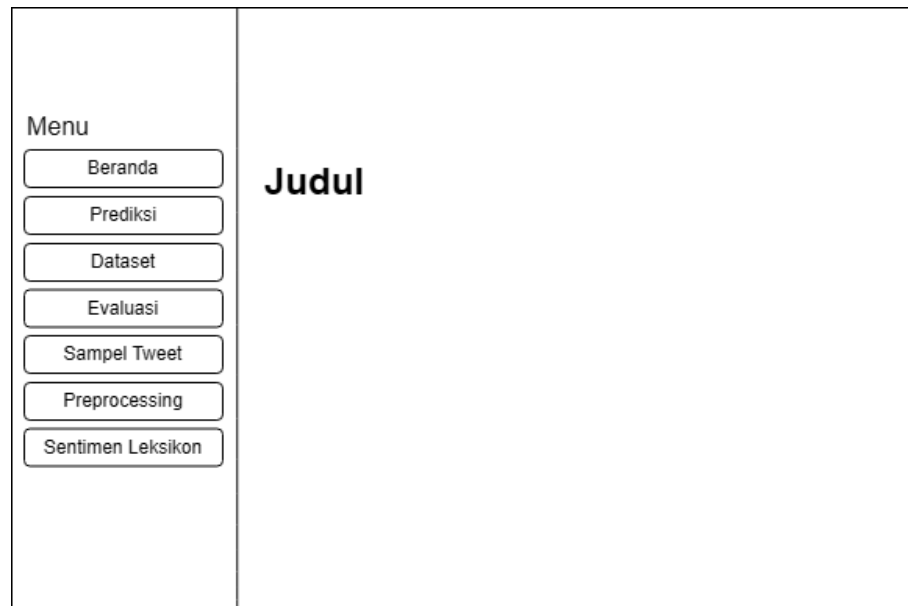
Gambar 3.14 Perancangan Proses

3. Perancangan *Interface*

Rancangan *interface* atau antarmuka pengguna adalah proses desain yang berfokus pada pengembangan model komunikasi antara pengguna dan sistem. Rancangan ini mencakup berbagai aspek, seperti tata letak dan navigasi, agar pengguna dapat berinteraksi dengan sistem secara mudah dan efisien. Terdapat satu aktor dalam sistem ini, yaitu *user* atau pengguna. Berikut ini merupakan gambaran rancangan antarmuka pengguna yang terbagi ke dalam beberapa menu sebagai berikut:

a. Rancangan Halaman Beranda

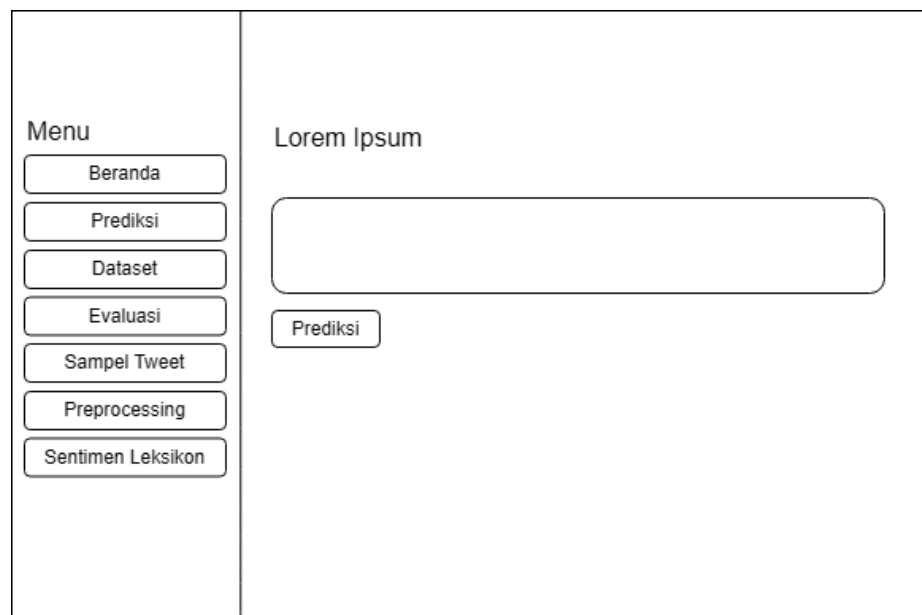
Halaman ini merupakan halaman awal sistem, dapat dilihat pada gambar 3.15 berikut:



Gambar 3.15 Rancangan Halaman Beranda

b. Rancangan Halaman Prediksi

Halaman ini digunakan untuk melakukan prediksi sentimen dari *input* yang diberikan oleh *user* atau pengguna, dapat dilihat pada gambar 3.16 berikut:



Gambar 3.16 Rancangan Halaman Prediksi Sentimen

c. Rancangan Halaman *Dataset*

Halaman *dataset* memungkinkan pengguna untuk melihat hasil dari setiap langkah yang dilewati *dataset*, dari mulai *dataset* yang masih mentah sampai ke *dataset* yang sudah dilakukan pelabelan sentimen. Antarmuka halaman *dataset* dapat dilihat pada gambar 3.17.

<div>Menu</div> <div>Beranda</div> <div>Prediksi</div> <div>Dataset</div> <div>Evaluasi</div> <div>Sampel Tweet</div> <div>Preprocessing</div> <div>Sentimen Leksikon</div>	Lorem Ipsum		
	Tabel		

Gambar 3.17 Rancangan Halaman *Dataset*

d. Rancangan Halaman Evaluasi

Halaman ini menampilkan hasil evaluasi model *KNN* dari parameter yang bisa ditentukan di dalam halaman.

Menu

Beranda

Prediksi

Dataset

Evaluasi

Sampel Tweet

Preprocessing

Sentimen Leksikon

Lorem Ipsum

Parameter

Metriks

Akurasi

Presisi

Recall

Confusion Matrix

Gambar 3.18 Rancangan Halaman Evaluasi

e. Rancangan Halaman Sampel *Tweet*

Menu ini memungkinkan pengguna untuk dapat mengambil kurang lebih 5 sampel data unggahan teks dari media sosial *X* sesuai dengan kueri atau kata kunci yang dimasukkan pengguna pada kolom *text box* yang tersedia.

<div>Menu</div> <div><div>Beranda</div><div>Prediksi</div><div>Dataset</div><div>Evaluasi</div><div>Sampel Tweet</div><div>Preprocessing</div><div>Sentimen Leksikon</div></div>	<div> Lorem Ipsum</div> <div><div></div></div> <div><div>Fetch</div></div>
--	--

Gambar 3.19 Rancangan Halaman Sampel *Tweet*

f. Rancangan Halaman *Preprocessing*

Halaman ini berfungsi sebagai simulasi tahapan-tahapan *preprocessing* data teks dengan cara memberikan *input* pada kolom *text box* yang tersedia.

<div>Menu</div> <div><div>Beranda</div><div>Prediksi</div><div>Dataset</div><div>Evaluasi</div><div>Sampel Tweet</div><div>Preprocessing</div><div>Sentimen Leksikon</div></div>	<div> Lorem Ipsum</div> <div><div></div></div> <div><div>Proses</div></div>
--	---

Gambar 3.20 Rancangan Halaman *Preprocessing*

g. Rancangan Halaman Sentimen Leksikon

Halaman ini berfungsi sebagai simulasi pelabelan sentimen suatu kalimat dengan cara memberikan *input* pada kolom *text box* yang tersedia.

<p>Menu</p> <p>Beranda</p> <p>Prediksi</p> <p>Dataset</p> <p>Evaluasi</p> <p>Sampel Tweet</p> <p>Preprocessing</p> <p>Sentimen Leksikon</p>	<p>Lorem Ipsum</p> <p></p> <p>Proses</p>
---	--

Gambar 3.21 Rancangan Halaman Sentimen Leksikon

3.2.3 Implementation

Tahapan ini merupakan fase ketika desain dan perancangan sistem yang telah dibuat akan direalisasikan menjadi kode program untuk menghasilkan sebuah sistem yang berfungsi. Sistem analisis sentimen ini diimplementasikan dan dirancang sesuai dengan metode yang digunakan berdasarkan tahapan analisis sentimen untuk memprediksi atau klasifikasi sentimen publik terhadap pemindahan ibu kota negara ke dalam tiga kelas yaitu positif, netral, dan negatif.

3.2.4 System Testing

Tahapan akhir ini merupakan tahapan untuk pengujian sistem. Pengujian ini bertujuan untuk menguji rancangan sistem yang sudah dibangun sebelumnya agar dapat melakukan evaluasi dari model yang digunakan untuk proses klasifikasi yang sudah di jalankan. *Black box* merupakan metode yang digunakan untuk tahapan *system testing* ini.

Metode *black box testing* memiliki tujuan untuk mengetahui efektivitas, keberhasilan serta kemampuan kerja sistem yang dikembangkan. Selain itu juga digunakannya metode *black box* bertujuan untuk memastikan fungsi setiap fitur yang terdapat pada sistem yang sudah dibangun sesuai dengan yang seharusnya. Berikut detail dari hasil pengujian sistem menggunakan metode *black box testing*:

Tabel 3.29 Rancangan Pengujian *Black Box*

Halaman	Detail Pengujian	Berhasil	Tidak Berhasil
<i>Dataset</i>	Menampilkan perubahan <i>dataset</i> dari setiap proses		
Evaluasi	Menampilkan informasi hasil dari evaluasi model masing-masing parameter		
Prediksi	Menampilkan prediksi sentimen dari <i>input</i> yang diberikan		
Sampel <i>Tweet</i>	Menampilkan 5 <i>tweets</i> berdasarkan kueri pencarian yang diberikan		
<i>Preprocessing</i>	Menampilkan hasil <i>preprocessing</i> dari <i>input</i> yang diberikan		
Sentimen Leksikon	Menampilkan hasil pelabelan sentimen <i>InSet</i> dari <i>input</i> yang diberikan		

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

Bagian ini berisi pengimplementasian dalam penelitian yang akan menguraikan secara rinci tentang penerapan rancangan yang telah disusun sebelumnya, serta menjelaskan langkah-langkah implementasi yang dilakukan dalam membangun model klasifikasi. Selain itu, akan dibahas juga mengenai proses pengembangan sistem yang digunakan dalam penelitian, yang mencakup bagaimana sistem dibangun dan diintegrasikan untuk mendukung analisis sentimen publik terhadap pemindahan ibu kota negara. Penjelasan akan mencakup penggunaan metode *Lexicon Based* untuk pelabelan sentimen, penerapan model *KNN* untuk klasifikasi, serta pengujian dan pengembangan sistem untuk memastikan kinerja yang optimal dalam proses analisis data.

4.1.1 Implementasi Model Klasifikasi

Pada bagian ini, akan dijelaskan hasil dari implementasi model klasifikasi yang telah dibangun berdasarkan rancangan yang telah disusun sebelumnya. Pembahasan akan difokuskan pada algoritma yang diterapkan dalam proses klasifikasi, khususnya penggunaan metode *Lexicon Based* untuk pelabelan sentimen dan model *K-Nearest Neighbor* (*KNN*) yang digunakan untuk klasifikasi sentimen publik terhadap pemindahan ibu kota negara. Selain itu, bagian ini akan menguraikan bagaimana kedua metode tersebut diintegrasikan untuk menghasilkan model yang dapat mengidentifikasi sentimen positif, netral, dan negatif dengan akurasi yang optimal.

a. Pengumpulan Data

Proses ini dilakukan dengan menggunakan teknik *web scraping* pada unggahan teks di media sosial *X (Twitter)* dari bulan Januari 2024 hingga Oktober 2024. Data diambil dengan bantuan *library TwiKit* yang ada pada lingkungan bahasa pemrograman *python* dan selanjutnya disimpan dalam format file *.csv*. Berikut merupakan modul program yang digunakan dalam proses *web scraping* dapat dilihat pada Algoritma 1.

Algoritma 1: web scraping

Input: twitter_auth_token, search_query

Output: file (.csv)

Function get_tweets(tweets)

 if tweets is None:

 print(f'{datetime.now()} - Mengambil tweet...')

 tweets = await client.search_tweet(CURRENT_QUERY,
product="Latest")

 else:

 wait_time = randint(5, 10)

 print(f'{datetime.now()} - Mengambil tweet
selanjutnya setelah {wait_time} detik...')

 time.sleep(wait_time)

 tweets = await tweets.next()

 return tweets

end function

b. *Preprocessing Data*

Setelah data berhasil diperoleh, data tersebut akan melalui tahap *preprocessing* untuk membersihkan dan meningkatkan kualitas data, agar siap digunakan dalam analisis lebih lanjut. Tahap awal dari *preprocessing* ini mencakup *import* data dengan menggunakan fungsi *read_csv()* yang terdapat pada *library pandas* yang berfungsi untuk membaca berkas data yang telah disimpan sebelumnya. Berikut merupakan modul program yang digunakan dalam proses *import dataset* dapat dilihat pada Algoritma 2.

Algoritma 2: *import dataset*

```
Input: file_path
Output: df (pandas DataFrame)
Function import_dataset(file_path)
DATASET_FILE_NAME = "merged_dataset.csv"
DATASET_FILE_PATH= f"datasets/merged/{DATASET_FILE_NAME}"
df = pd.read_csv(DATASET_FILE_PATH)
return df
end function
```

1. *Text Cleaning*

Setelah *import* data berhasil, langkah selanjutnya adalah proses *text cleaning* yang bertujuan untuk menghilangkan elemen-elemen yang tidak dibutuhkan pada sebuah dokumen guna memastikan data yang digunakan lebih bersih dan siap untuk analisis lebih lanjut. Berikut merupakan modul program yang digunakan dalam proses *text cleaning* dapat dilihat pada Algoritma 3.

Algoritma 3: *text cleaning*

```
Input: df (pandas DataFrame)
Output: df (pandas DataFrame)
Function clean_text(text)
    text = re.sub(r'RT\s', '', text)
    text = re.sub(r'\@([\w]+)', " ", text)
    text = re.sub(r'\\u[a-zA-Z0-9]{4}', " ", text)
    text = re.sub(r'\n\s', " ", text)
    text = re.sub(r'\n', " ", text)
    text = re.sub(r'^\x00-\x7F+', '', text)
    text = re.sub(r'([a-zA-Z])\1\1','\1', text)
    text = re.sub(r'http[s]?\:\/\/\.[a-zA-Z0-9\.\_\/\_]?=%&#\-\+!]+', '', text)
    text = re.sub(r'pic.twitter.com?.[a-zA-Z0-9\.\_\/\_]?=%&#\-\+!]+', '', text)
    text = text.lower()
    text = re.sub(r'\#[a-zA-Z0-9_]+', '', text)
    ...
    text = re.sub(r'^[ ]|[ ]$', '', text)
return text
end function
```

2. Pembakuan Kata

Proses ini mengganti kata slang atau gaul menjadi kata baku dengan menggunakan kamus kata slang yang didapatkan dari beberapa sumber di internet agar model dapat lebih akurat dalam melakukan klasifikasi sentimen. Berikut merupakan modul program yang digunakan dalam proses pembakuan kata dapat dilihat pada Algoritma 4.

Algoritma 4: Pembakuan Kata

```
Input: df (pandas DataFrame)
Output: df (pandas DataFrame)
Function standardize_text(text)
    words = text.split()
    standardization_words = []

    for word in words:
        if word in SLANG_DICTIONARY_1:
            standardization_words.append(SLANG_DICTIONARY_1[
word])
        else:
            standardization_words.append(word)

    return text
end function
```

3. Negation Handling

Selanjutnya, untuk menangani kata-kata yang memiliki pengaruh negatif dalam analisis sentimen, dilakukan langkah *negation handling*. Proses ini mengubah bentuk kata yang memiliki makna yang sama tanpa kata negasi seperti “tidak” dan “belum” yang dapat mengubah makna kalimat dan perlu dipertimbangkan secara khusus. Berikut merupakan modul program yang digunakan dalam proses *negation handling* dapat dilihat pada Algoritma 5.

Algoritma 5: *negation handling*

```
Input: df (pandas DataFrame)
Output: df (pandas DataFrame)
Function swap_antonyms(text)
    words = text.split()
    antonym_dict=dict(zip(df_dict["word"],df_dict["antonim"])
        new_words = []
        for word in words:
            if "_" in word:
                negation, next_word = word.split("_", 1)
                if next_word in antonym_dict:
                    new_words.append(antonym_dict[next_word])
                else:
                    new_words.append(word)
            else:
                new_words.append(word)

    return text
end function
```

4. *Stopwords Removal*

Proses berikutnya adalah *stopwords removal*, bertujuan untuk menghilangkan kata yang dianggap tidak memiliki makna signifikan atau tidak relevan yang terdapat pada dokumen. Contoh *stopwords* yang ada pada Bahasa Indonesia seperti “yang”, “dan”, “di”, “itu”, “adapun”, “agak” dan sebagainya. Langkah ini bertujuan untuk mengurangi dimensi data dan meningkatkan efisiensi model dalam menganalisis sentimen. Berikut merupakan modul program yang digunakan dalam proses *stopwords removal* dapat dilihat pada Algoritma 6.

Algoritma 6: *stopwords removal*

```
Input: df (pandas DataFrame)
Output: df (pandas DataFrame)
Function drop_stopwords(text)
    custom_stopwords = stopwords.words('indonesian')
    custom_stopwords.clear()
    custom_stopwords.extend(DATA_FRAME_SW_DICTIONARY
                             _1["stopwords"].values)

    factory = StopWordRemoverFactory()
    sastrawi_stopwords = factory.get_stop_words()

    combined_stopwords =
    set(custom_stopwords).union(set(sastrawi_stopwords))

    return " ".join([word for word in text.split() if word
                      not in combined_stopwords])

end function
```

5. *Stemming*

Proses ini bertujuan untuk mengubah kata-kata turunan menjadi bentuk dasarnya, misalnya "berjalan" menjadi "jalan". *Stemming* membantu mengurangi variasi kata dan membuat model lebih fokus pada makna inti dari kata tersebut. Berikut merupakan modul program yang digunakan dalam proses *stemming* dapat dilihat pada Algoritma 7.

Algoritma 7: *stemming*

```
Input: df (pandas DataFrame)
Output: df (pandas DataFrame)
Function stem_text(text)
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()

    return " ".join([stemmer.stem(word) for word in
                      text.split()])

end function
```

c. Pelabelan *Lexicon Based*

Setelah *preprocessing* selesai, data yang telah diproses akan melalui tahap pelabelan data, dimana setiap data akan diberi label berdasarkan aspek sentimen yang terkandung dalam teks tersebut. Pelabelan ini dilakukan menggunakan kamus *lexicon* yang berisi daftar kata-kata beserta nilai sentimennya. Berdasarkan analisis kata-kata dalam *tweet*, setiap teks akan diberi label sesuai dengan sentimen yang tercermin. *Tweet* dengan nilai lebih dari 0 akan diberi label positif, yang bernilai kurang dari 0 akan diberi label negatif, dan yang bernilai sama dengan 0 akan diberi label netral. Berikut merupakan modul program yang digunakan dalam proses pelabelan data dapat dilihat pada Algoritma 8.

Algoritma 8: *Lexicon Based*

```
with open("./lexicons/lexicon_json_inset-neg.txt") as f:
    inset_neg = f.read()
with open("./lexicons/lexicon_json_inset-pos.txt") as f:
    inset_pos = f.read()

insetNeg = json.loads(inset_neg)
insetPos = json.loads(inset_pos)

# inisiasi vader sentiment analyzer
sia_inset_neg = SentimentIntensityAnalyzer()
sia_inset_pos = SentimentIntensityAnalyzer()

# clear default lexicon
sia_inset_neg.lexicon.clear()
sia_inset_pos.lexicon.clear()

# assign InSet lexicon
sia_inset_neg.lexicon.update(insetNeg)
sia_inset_pos.lexicon.update(insetPos)

def get_inset_compound_score(text):
    inset_compound_score=
    sia_inset_neg.polarity_scores(text) ["compound"]+
    sia_inset_pos.polarity_scores(text) ["compound"]
    return inset_compound_score

def get_inset_label(number):
    if number < 0:
        return "negatif"
    elif number > 0:
        return "positif"
    else:
        return "netral"
```

d. Vektorisasi *Term Frequency – Inverse Document Frequency (TF-IDF)*

Langkah selanjutnya adalah penerapan proses pembobotan *TF-IDF (Term Frequency-Inverse Document Frequency)* yang bertujuan untuk memberikan bobot pada setiap istilah atau *term* dalam data teks. Pembobotan ini dilakukan dengan menghitung nilai *Term Frequency (TF)* yang menggambarkan frekuensi kemunculan suatu kata dalam dokumen. Proses pembobotan ini bertujuan untuk memberikan bobot yang lebih tinggi pada kata-kata yang muncul secara spesifik dalam dokumen, tetapi jarang muncul dalam dokumen lain, sehingga dapat memberikan kontribusi yang lebih besar pada model. Modul pembobotan *TF-IDF* ini menggunakan fungsi *TfidfVectorizer()* yang berfungsi untuk menghitung dan mengaplikasikan nilai-nilai *TF-IDF* pada setiap *term* dalam *dataset* yang digunakan, menghasilkan representasi vektor yang lebih informatif untuk analisis selanjutnya. Berikut merupakan modul program yang digunakan dalam proses vektorisasi *TF-IDF* dapat dilihat pada Algoritma 9.

Algoritma 9: *TF-IDF*

```
tfidf = TfidfVectorizer()  
X_tfidf=tfidf.fit_transform(sentiment["preprocessed_text  
"]).toarray()
```

e. *Oversampling SMOTE*

Proses berikutnya adalah penerapan *resampling* yaitu *oversampling* dengan *SMOTE* yang berguna untuk mengatasi ketidakseimbangan kelas dalam *dataset*. Teknik ini fokus pada kelas minoritas dalam *dataset*, dimana jumlah sampelnya jauh lebih sedikit dibandingkan dengan kelas mayoritas. Berikut merupakan modul program yang digunakan dalam proses *oversampling* dengan *SMOTE* dapat dilihat pada Algoritma 10.

Algoritma 10: *SMOTE*

```
smote = SMOTE(random_state=21)  
X_smote, Y_smote = smote.fit_resample(X_tfidf,  
sentiment["sentiment_label"])
```

f. Klasifikasi *K-Nearest Neighbor (KNN)*

Proses klasifikasi dilakukan dengan menggunakan model *K-Nearest Neighbor (KNN)*. Langkah pertama adalah memuat data yang telah melalui proses *oversampling* dengan *SMOTE*. Selanjutnya, data tersebut digunakan sebagai target untuk pelatihan model. Proses pembagian data dilakukan dengan membagi 80% data pelatihan dan 20% data pengujian. Model *K-Nearest Neighbor (KNN)* kemudian dilatih menggunakan data latih yang telah dipersiapkan. Setelah model selesai dilatih, model ini digunakan untuk memprediksi label sentimen pada data uji yang telah disiapkan. Berikut merupakan modul program pelatihan model algoritma klasifikasi *K-Nearest Neighbor* dapat dilihat pada Algoritma 11.

Algoritma 11: KNN

```
def train_evaluate_knn(k, X_train, Y_train, X_test,
Y_test, ratio, w, m):
    knn = KNeighborsClassifier(n_neighbors=k, weights=w,
metric=m)
    knn.fit(X_train, Y_train)
    y_pred = knn.predict(X_test)

    accuracy = accuracy_score(Y_test, y_pred)
    precision = precision_score(Y_test, y_pred,
average='weighted')
    recall = recall_score(Y_test, y_pred,
average='weighted')
    f1 = f1_score(Y_test, y_pred, average='weighted')
    cm = confusion_matrix(Y_test, y_pred,
labels=knn.classes_)
    disp = ConfusionMatrixDisplay(confusion_matrix=cm,
display_labels=knn.classes_)
    disp.plot()
    plt.title(f"confusion matrix for k={k}, {ratio}
split")
    plt.show()

    return knn

best_9010 = gs_9010.best_params_
best_8020 = gs_8020.best_params_
best_7030 = gs_7030.best_params_

knn_9010 = train_evaluate_knn(best_9010["n_neighbors"],
X_train_9010, Y_train_9010, X_test_9010, Y_test_9010,
"90:10", best_9010["weights"], best_9010["metric"])
knn_8020 = train_evaluate_knn(best_8020["n_neighbors"],
X_train_8020, Y_train_8020, X_test_8020, Y_test_8020,
"80:20", best_8020["weights"], best_8020["metric"])
knn_7030 = train_evaluate_knn(best_7030["n_neighbors"],
X_train_7030, Y_train_7030, X_test_7030, Y_test_7030,
"70:30", best_7030["weights"], best_7030["metric"])
```

g. *Tuning Hyper Parameter GridSearch*

Proses ini merupakan tahapan untuk melakukan pencarian parameter terbaik untuk sebagai argumen yang akan dimasukkan ketika menjalankan sumber kode pelatihan model *KNN* yang mana bertujuan untuk mencari hasil akurasi yang paling baik. Beberapa parameter akan diinisialisasikan terlebih dahulu sebagai skenario yang akan diuji. Berikut merupakan modul program inisiasi parameter dengan algoritma *GridSearch* dapat dilihat pada Algoritma 12.

Algoritma 12: *GridSearch*

```
gs_knn = KNeighborsClassifier()
param_grid = {
    'n_neighbors': [1, 3, 5, 7, 9],
    'weights': ['distance'],
    'metric': ['euclidean', 'manhattan', 'cosine']
}

def get_grid_search(X_train, Y_train, X_test, Y_test,
ratio):
    # Initialize GridSearchCV with your model, parameter
    grid, and number of cross-validation folds
    grid_search = GridSearchCV(estimator=gs_knn,
param_grid=param_grid, cv=10, scoring='accuracy',
n_jobs=-1)

    # Assuming you have your training data X_train and
y_train
    grid_search.fit(X_train, Y_train)

    # Best model
    best_knn = grid_search.best_estimator_

print("=====
=====")
    print(f"| {ratio} split |")

print("=====
=====")
    # Best hyperparameters
    print("Best Parameters:", grid_search.best_params_)
    # Evaluate the best model on your test set
    y_pred = best_knn.predict(X_test)
    accuracy = accuracy_score(Y_test, y_pred)
    print("Test Set Accuracy:", accuracy)

    return grid_search

gs_9010 = get_grid_search(X_train_9010, Y_train_9010,
X_test_9010, Y_test_9010, "90:10")
gs_8020 = get_grid_search(X_train_8020, Y_train_8020,
X_test_8020, Y_test_8020, "80:20")
gs_7030 = get_grid_search(X_train_7030, Y_train_7030,
X_test_7030, Y_test_7030, "70:30")
```

4.1.2 Implementasi Pengembangan Sistem

Implementasi pengembangan sistem pada penelitian ini dilakukan dengan menggunakan *library streamlit* yang memungkinkan pembuatan antarmuka pengguna (*user interface*) yang interaktif dan fungsional. Pada tahap ini, hasil dari desain sistem yang telah direncanakan sebelumnya diwujudkan dalam bentuk aplikasi *web* yang dapat diakses dan digunakan langsung oleh pengguna. Desain sistem tidak hanya mencakup tampilan visual dari antarmuka, tetapi juga menyertakan berbagai fungsi yang memungkinkan pengguna berinteraksi dengan sistem secara efektif dan efisien. Setiap halaman yang telah dirancang menampilkan bagaimana sistem ini berfungsi dan bekerja sesuai dengan spesifikasi yang telah ditetapkan pada tahap perancangan. Setiap elemen antarmuka dan fungsi yang telah direncanakan dapat diimplementasikan dengan mudah dan cepat, memberikan gambaran yang jelas tentang bagaimana aplikasi ini beroperasi dan berfungsi secara keseluruhan.

a. Halaman Beranda

Halaman ini merupakan tampilan pertama yang muncul saat sistem dijalankan. Halaman beranda menampilkan judul penelitian secara jelas. Tampilan lengkap dari halaman beranda ini dapat dilihat pada gambar 4.1 berikut:



Gambar 4.1 Halaman Beranda

b. Halaman Prediksi Sentimen

Menu ini dirancang untuk memungkinkan pengguna melakukan *input tweet* berupa teks yang ingin diuji untuk analisis sentimennya. Pengguna dapat langsung memasukkan *input* yang relevan dan sistem akan otomatis menganalisis sentimen yang terkandung dalam *input* tersebut serta mengidentifikasi kategori sentimen yang terkait dengan konten tersebut. Halaman prediksi sentimen dapat dilihat pada gambar 4.2 berikut:

Masukkan kalimat terkait topik relokasi ibu kota dan dapatkan prediksi sentimennya

Masukkan kalimat di sini:

Prediksi Sentimen

Gambar 4.2 Halaman Prediksi Sentimen

c. Halaman *Dataset*

Menu *dataset* menampilkan beberapa proses data yang digunakan pada penelitian ini. Pengguna dapat melihat data dalam bentuk aslinya sebelum diproses lebih lanjut.

1. *Dataset* Mentah

Berikut merupakan tampilan *dataset* hasil dari *web scraping* dapat dilihat pada gambar 4.3.

Fresh Dataset							
	Unnamed: 0	no	urls	user_id	username	user_display_name	tweet_id
0	0	1	[]	⚠ 1259337235469500418	amourafolie	CATALINKA TATJANA.	⚠ 1850
1	1	2	[]	⚠ 1310837194332880898	nornoranao	Nora 論文時代	⚠ 1850
2	2	3	[]	⚠ 1655455881784623104	pixelplay808	노을 소나기 🌧️ 🌧️ 🌧️	⚠ 1850
3	3	4	[{'displ	154,102,750	Beritasatu	BeritaSatu	⚠ 1850
4	4	5	[]	1,647,300,211	wprimandika	wanda.	⚠ 1850
5	5	6	[]	⚠ 1259337235469500418	amourafolie	CATALINKA TATJANA.	⚠ 1850
6	6	7	[]	⚠ 1752174695468953600	PamanMarco	paman marco	⚠ 1850
7	7	8	[]	2,299,524,572	SobatHAPE	SobatHAPE	⚠ 1850
8	8	9	[]	⚠ 1649241151625650176	louielouiesky	louie	⚠ 1850
9	9	10	[]	⚠ 1648990500446425091	popapotiee	🌟🌟	⚠ 1850

Gambar 4.3 Tampilan *Dataset* Mentah

2. *Dataset* Setelah *Preprocessing*

Berikut merupakan tampilan *dataset* hasil dari *preprocessing* dapat dilihat pada gambar 4.4.

	after_stemming_text	preprocessed_text
19	tung banding pindah ibukota	tung banding pindah ibukota
20	tonton sandiwara mk lolos anak periode pindah ibu	tonton sandiwara mk lolos anak periode pindah ibu
21	gara bang rungkad argumen historis pindah ibukota	gara bang rungkad argumen historis pindah ibukota
22	simak ustadz alpian tanjung pki turun oligarki cina	simak ustadz alpian tanjung pki turun oligarki cina
23	buku sukarso pindah ibukota isi atur iya buat isu bo	buku sukarso pindah ibukota isi atur iya buat isu bo
24	ide pindah ibukota layak juang langkah ukur bijaksa	ide pindah ibukota layak juang langkah ukur bijaksa
25	mikir pindah kota direncanaiin matang matang sih	mikir pindah kota direncanaiin matang matang sih
26	pindah ibukota gagas zaman jaya samudra pasai	pindah ibukota gagas zaman jaya samudra pasai
27	ibukota negara anggap strategis bang iya tulis pinda	ibukota negara anggap strategis bang iya tulis pinda
28	deh facto jakarta pusat perintah negara republik inc	deh facto jakarta pusat perintah negara republik inc

Gambar 4.4 Tampilan *Dataset* Setelah *Preprocessing*

3. *Dataset* Setelah Pelabelan Sentimen

Berikut merupakan tampilan *dataset* hasil dari proses pelabelan sentimen dapat dilihat pada gambar 4.5.

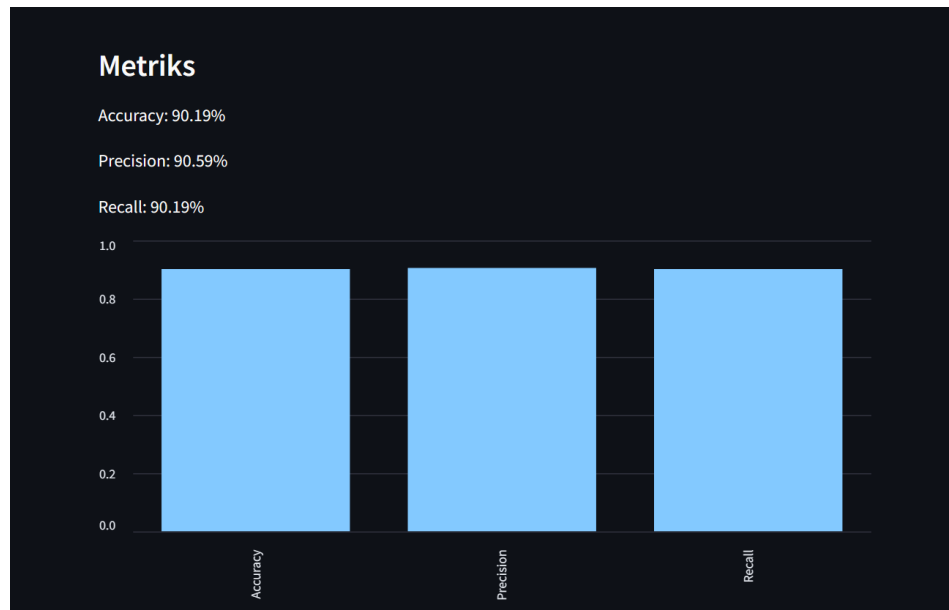
	negative_score	sentiment_label
0 s': 0.0, 'compound': 0.0}	{'neg': 0.571, 'neu': 0.429, 'pos': 0.0, 'compound': -0.6124}	negatif
1 pos': 0.667, 'compound': 0.6124}	{'neg': 0.667, 'neu': 0.333, 'pos': 0.0, 'compound': -0.6124}	netral
2 s': 1.0, 'compound': 0.4588}	{'neg': 1.0, 'neu': 0.0, 'pos': 0.0, 'compound': -0.25}	positif
3 s': 0.0, 'compound': 0.0}	{'neg': 0.545, 'neu': 0.455, 'pos': 0.0, 'compound': -0.7906}	negatif
4 s': 0.0, 'compound': 0.0}	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}	netral
5 s': 0.0, 'compound': 0.0}	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}	netral
6 s': 0.0, 'compound': 0.0}	{'neg': 1.0, 'neu': 0.0, 'pos': 0.0, 'compound': -0.9953}	negatif
7 s': 0.0, 'compound': 0.0}	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}	netral
8 s': 0.0, 'compound': 0.0}	{'neg': 0.571, 'neu': 0.429, 'pos': 0.0, 'compound': -0.4588}	negatif
9 s': 0.0, 'compound': 0.0}	{'neg': 0.333, 'neu': 0.667, 'pos': 0.0, 'compound': -0.25}	negatif

Gambar 4.5 Tampilan *Dataset* Setelah Pelabelan Sentimen

d. Halaman Evaluasi Model

Halaman ini dirancang untuk menampilkan hasil evaluasi model klasifikasi yang telah dilatih menggunakan data uji. Pengguna dapat melihat metrik performa model seperti akurasi, *precision*, dan *recall* yang menunjukkan sejauh mana model dapat melakukan klasifikasi sentimen dengan benar. Setiap metrik ini dihitung berdasarkan hasil prediksi yang dibandingkan dengan label sebenarnya dalam data uji. Halaman ini juga menyertakan visualisasi berupa grafik yang mempermudah pengguna untuk memahami performa model secara lebih jelas dan mendalam. Selain itu, evaluasi juga mencakup analisis kesalahan dan saran perbaikan untuk model agar lebih optimal di masa depan. Hasil evaluasi ini memberikan gambaran

lengkap mengenai kinerja sistem dalam melakukan analisis sentimen dari dari teks dapat dilihat pada gambar 4.6 berikut.



Gambar 4.6 Halaman Evaluasi Model

e. Halaman Sampel *Tweet*

Menu ini memungkinkan pengguna untuk dapat mengambil kurang lebih 5 sampel data unggahan teks dari media sosial *X* sesuai dengan kueri atau kata kunci yang dimasukkan pengguna pada kolom *text box* yang tersedia. Halaman sampel *tweet* dapat dilihat pada gambar 4.7 berikut.

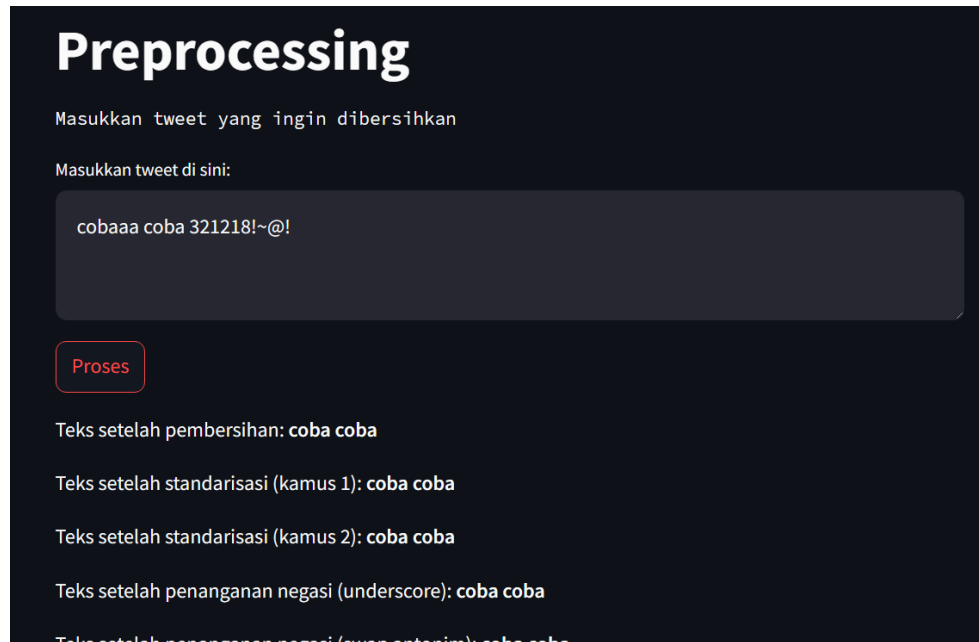
The figure shows a web interface titled "Sampel Tweet" (Sample Tweet) on a dark background. Below the title, it says "Scraping sampel tweet". There is a section labeled "Masukkan query pencarian twitter:" (Enter Twitter search query:) with a text input field containing the word "test". Below the input field is a red-outlined button labeled "Fetch". Below the button is a table displaying 5 sample tweets. The table has 7 columns: an index column, a "#" column, "tweet_id", "user_id", "username", "display_name", and "text".

	#	tweet_id	user_id	username	display_name	text
0	1	1877930556266090544	1709772420989456386	Spartacus_test	Rafa Spartano	@yaraadeusa
1	2	1877930551367377076	1799156630589939713	DTDavisPhD	DTDavis	@stkirsch Unk
2	3	1877930539996373490	1142141265544265728	dry_c0la	🌙 Dry Cola ☀️	Hey is anyone
3	4	1877930514067210240	1009924943977558016	JustAWeirdOwl	Imaginary	Love my new \$
4	5	1877930507163594849	713842195904770048	Ethiereall	yuin	Test #zonauan

Gambar 4.7 Halaman Sampel *Tweet*

f. Halaman *Preprocessing*

Halaman ini berfungsi sebagai simulasi tahapan-tahapan *preprocessing* data teks dengan cara memberikan *input* pada kolom *text box* yang tersedia dapat dilihat pada gambar 4.8 berikut.



The screenshot shows a web application titled "Preprocessing". It has a dark background with white text. At the top, it says "Masukkan tweet yang ingin dibersihkan" (Enter the tweet you want to clean). Below that, it says "Masukkan tweet di sini:" (Enter the tweet here:). There is a text input box containing the text "cobaaa coba 321218!~@!". Below the input box is a red button labeled "Proses". Under the button, there are four lines of text showing the results of the preprocessing steps: "Teks setelah pembersihan: coba coba", "Teks setelah standarisasi (kamus 1): coba coba", "Teks setelah standarisasi (kamus 2): coba coba", and "Teks setelah penanganan negasi (underscore): coba coba".

Gambar 4.8 Halaman *Preprocessing*

g. Halaman Sentimen Leksikon

Halaman ini berfungsi sebagai simulasi pelabelan sentimen suatu kalimat dengan cara memberikan *input* pada kolom *text box* yang tersedia dapat dilihat pada gambar 4.9 berikut.



The screenshot shows a web application titled "Label Sentimen Lexicon (InSet)". It has a dark background with white text. At the top, it says "Masukkan kalimat yang ingin dilabeli sentimennya" (Enter the sentence you want to label its sentiment). Below that, it says "Masukkan kalimat di sini:" (Enter the sentence here:). There is a text input box containing the text "coba inset". Below the input box is a red button labeled "Proses". Under the button, there is a JSON object representing the sentiment label:

```
{
  "sentiment": "positif",
  "compound_score": 0.20879999999999999,
  "positive_score": {
    "neg": 0
  }
}
```

Gambar 4.9 Halaman Sentimen Leksikon

4.2 Hasil

Bagian ini berisi penguraian secara rinci temuan dari seluruh proses pengujian yang telah dilakukan mencakup analisis hasil pengujian model klasifikasi untuk mengukur kinerja akurasi, *precision*, *recall*, dan metrik evaluasi lainnya, serta pengujian terhadap pengembangan sistem yang diterapkan dalam analisis sentimen publik terhadap pemindahan ibu kota negara. Uraian hasil pengujian ini bertujuan untuk memberikan pemahaman mendalam tentang seberapa baik model dan sistem yang dikembangkan dalam memenuhi tujuan penelitian serta efektivitas metode yang digunakan.

4.2.1 Proses *Training Model*

Bagian ini akan menjelaskan secara rinci mengenai pengujian model klasifikasi yang diterapkan dalam penelitian ini, termasuk hasil-hasil yang diperoleh. Pengujian dilakukan untuk mengevaluasi sejauh mana model dapat memprediksi dengan akurat sentimen dari data unggahan *tweet* yang diberikan. Sebagai penilaian kinerja model klasifikasi, digunakan metode *confusion matrix*, yang memberikan analisis menyeluruh tentang performa model dalam melakukan klasifikasi data.

a. *Labelling Lexicon Based*

Berikut merupakan hasil persebaran kelas berdasarkan *labelling* menggunakan *lexicon based* dapat dilihat pada tabel 4.1.

Tabel 4.1 Persebaran Kelas

Sentimen	Jumlah
Negatif	3039
Positif	1341
Netral	533

Berdasarkan tabel 4.1 di atas dapat dilihat bahwa kelas sentimen negatif memiliki jumlah data atau label terbanyak, dimana proses pelabelan menggunakan *lexicon based* dengan kamus *InSet* menghasilkan 3039 data berlabel negatif atau 61.86% dari *dataset* yang ada. Sentimen dengan kelas positif didapatkan jumlah sebesar 1341 data atau 27.29% dari *dataset* yang ada. Terakhir pada kelas netral didapatkan jumlah 533 data dengan persentase 10.85% dari total data yang ada. Hal ini terjadi karena proses *web scraping* yang sebagian besar menghasilkan *tweet* yang bersifat tanggapan negatif setelah dilabeli dengan kamus leksikon *InSet* melalui metode *lexicon based*. Data yang digunakan dalam penelitian ini dibagi dengan rasio 80:20, dimana 80% data digunakan untuk proses pelatihan (*training*) dan 10% sisanya digunakan untuk pengujian (*testing*).

b. *Oversampling SMOTE*

Berdasarkan hasil proses sebelumnya, diketahui bahwa data tersebut terindikasi tidak seimbang (*imbalanced data*) dikarenakan perbedaan jumlah dari setiap kelas sangat jauh. Proses *SMOTE* menghasilkan persebaran data yang dapat dilihat pada tabel 4.2.

Tabel 4.2 Persebaran Kelas Setelah *SMOTE*

Sentimen	Jumlah
Negatif	3039
Positif	3039
Netral	3039

c. *Inisiasi Hyper Parameter*

Proses ini merupakan tahapan untuk mencari parameter terbaik untuk digunakan dalam proses pelatihan model. Adapun parameter yang akan digunakan dalam proses ini dapat dilihat pada tabel 4.3 berikut.

Tabel 4.3 Inisiasi Parameter

<i>K</i>	<i>Weights</i>	<i>Metric</i>
1	<i>distance</i>	<i>euclidean</i>
3	<i>uniform</i>	<i>manhattan</i>
5		<i>cosine</i>
7		
9		

d. *Training KNN*

Pelatihan model *KNN* pada penelitian ini akan menggunakan beberapa parameter untuk mencari performa terbaik dalam menghasilkan akurasi model. Selanjutnya, skenario dari beberapa parameter itu akan diuji untuk menemukan kombinasi parameter terbaik dengan melakukan inisiasi parameter *k*, *weights*, dan juga *metric* pada tabel 4.3. Hal ini bertujuan untuk mengetahui *hyper parameter* terbaik dalam melakukan evaluasi model ketika menghasilkan akurasi. Berikut merupakan hasil pelatihan model *KNN* dengan menggunakan tiga skenario rasio pembagian data dapat dilihat pada tabel 4.4.

Tabel 4.4 *Training KNN*

Rasio	Akurasi	<i>Precision</i>	<i>Recall</i>
90:10	89.80%	90.38%	89.80%
80:20	90.18%	90.59%	90.18%
70:30	88.66%	89.32%	88.66%

Tabel 4.4 atas menunjukkan bahwa pengujian masing-masing rasio pembagian data berhasil dilakukan. Pengujian tersebut menggunakan metode *gridsearch* yang mana metode tersebut merupakan salah satu cara yang bisa digunakan untuk mencari *hyper parameter* terbaik dalam proses pengujian model *KNN*.

4.2.2 Proses Pengujian Model

Bagian ini akan menjelaskan secara rinci mengenai pengujian model klasifikasi yang dikembangkan dalam penelitian ini, beserta hasil yang diperoleh. Pengujian dilakukan untuk mengevaluasi seberapa efektif model dalam memprediksi sentimen pada data unggahan *tweet*. Sebagai penilaian kinerja model, digunakan metode *confusion matrix* yang mana metode tersebut adalah hasil dari pengujian parameter terbaik dan digunakan *confusion matrix multiclass*, mengingat adanya tiga kelas sentimen yang perlu diklasifikasikan. Penggunaan *confusion matrix multiclass* ini memungkinkan evaluasi yang akurat terhadap kinerja model melalui metrik seperti akurasi, *precision*, *recall*, dan *f1-score*.

Pengujian ini bertujuan untuk mengukur performa model dengan menghasilkan nilai akurasi, *precision*, *recall*, dan *f1-score* untuk setiap kelas sentimen yaitu positif, negatif, dan netral. Pengujian awal dilakukan pada model *KNN* untuk klasifikasi kelas sentimen dapat dilihat pada tabel 4.5.

Tabel 4.5 Pengujian *Confusion Matrix*

		<i>Predicted</i>		
		-1 (<i>Negative</i>)	+1 (<i>Positive</i>)	0 (<i>Neutral</i>)
<i>Actual</i>	-1 (<i>Negative</i>)	462	61	78
	+1 (<i>Positive</i>)	26	556	13
	0 (<i>Neutral</i>)	1	0	627

Berdasarkan tabel 4.5 di atas terlihat hasil dari *confusion matrix* sentimen hasil pengujian model yang terbagi menjadi tiga kelas sentimen yaitu positif, negatif, dan netral. Berdasarkan tabel tersebut juga terdapat 462 data negatif yang benar terprediksi negatif, tetapi ada 61 data negatif terbaca positif dan 78 data negatif terbaca netral. Kemudian, pada kelas positif terdapat 26 data terbaca negatif, 13 data terbaca netral, dan 556 data terprediksi dengan benar. Terakhir, pada kelas netral terdapat 1 data yang terbaca negatif, tidak ada data yang terbaca positif, dan 627 data terprediksi benar. Hasil ini menunjukkan adanya beberapa kesalahan prediksi pada beberapa kategori kelas sentimen, terutama pada kelas negatif dan positif, yang mengindikasikan bahwa model masih mengalami kesulitan dalam membedakan beberapa kategori sentimen tertentu. Kalkulasi nilai akurasi, *precision*, *recall*, dan *f1-score* dari model *KNN* untuk kategori sentimen ini diperoleh berdasarkan dari hasil visualisasi *confusion matrix*. Berikut merupakan perhitungan akurasi, *precision*, *recall*, serta *f1-score*.

1. Penghitungan Akurasi

$$Accuracy = \frac{556+462+627}{462+61+78+26+556+13+1+0+627} = 0.9018 \times 100\% = 90.18\%$$

2. Penghitungan *Precision*

$$Preci\ Positive = \frac{556}{556+61+0} = 0.9011$$

$$Preci\ Negative = \frac{462}{462+26+1} = 0.9447$$

$$Preci\ Neutral = \frac{627}{627+78+13} = 0.8732$$

$$Precision = \frac{0.9011+0.9447+0.8732}{3} \times 100\% = 90.63\%$$

3. Penghitungan *Recall*

$$Recall\ Positive = \frac{556}{556+26+13} = 0.9344$$

$$Recall\ Negative = \frac{462}{462+61+78} = 0.7687$$

$$Recall\ Neutral = \frac{627}{627+1+0} = 0.9984$$

$$Recall = \frac{0.9344+0.7687+0.9984}{3} \times 100\% = 90.05\%$$

4. Penghitungan *F1-Score*

$$F1\ Positive = 2 \times \frac{0.9011 \times 0.9344}{0.9011+0.9344} = 0.9174$$

$$F1\ Negative = 2 \times \frac{0.9447 \times 0.7687}{0.9447+0.7687} = 0.8476$$

$$F1\ Neutral = 2 \times \frac{0.8732 \times 0.9984}{0.8732+0.9984} = 0.9316$$

$$F1\ Score = \frac{0.9174+0.8476+0.9316}{3} \times 100\% = 89.88\%$$

Berdasarkan hasil evaluasi, model klasifikasi sentimen mencapai akurasi sebesar 90.18%. Hal tersebut menunjukkan bahwa 90.18% dari total prediksi sesuai dengan data yang sebenarnya. Rata-rata nilai presisi mencapai 90.63%, mengindikasikan bahwa 90.63% dari prediksi model untuk kelas positif, netral, dan negatif adalah benar. Rata-rata *recall* sebesar 90.05%, model mampu mengidentifikasi 90.05% dari data aktual untuk setiap kelas sentimen. Secara spesifik, untuk kelas positif, presisi tercatat sebesar 0.9011, *recall* 0.9344, dan *f1-score* 0.9174. Kelas negatif memiliki presisi 0.9447, *recall* 0.7687, dan *f1-score* 0.8476. Sementara itu, kelas netral memperoleh hasil dengan presisi 0.8732, *recall* 0.9984, dan *f1-score* 0.9316. Rata-rata *f1-score* model sebesar 89.88% mencerminkan keseimbangan yang cukup baik antara presisi dan *recall*.

4.2.3 Pengujian Pengembangan Sistem

Pengujian sistem dalam penelitian ini dilakukan menggunakan metode *black-box testing* yang berfokus pada pengujian fungsionalitas sistem tanpa melihat implementasi internalnya. Pada pengujian ini, sistem diuji untuk memastikan kemampuannya dalam menjalankan proses klasifikasi teks terhadap data unggahan teks yang diperoleh dari media sosial *X (Twitter)*. Pengujian ini bertujuan untuk mengevaluasi sejauh mana sistem dapat melakukan analisis sentimen secara akurat. Hasil dari pengujian ini dapat dilihat pada tabel berikut.

Tabel 4.6 Hasil Pengujian *Black Box*

Halaman	Detail Pengujian	Hasil
<i>Dataset</i>	Menampilkan perubahan <i>dataset</i> dari setiap proses	Berhasil
Evaluasi	Menampilkan informasi hasil dari evaluasi model masing-masing parameter	Berhasil
Prediksi	Menampilkan prediksi sentimen dari <i>input</i> yang diberikan	Berhasil
Sampel <i>Tweet</i>	Menampilkan 5 <i>tweets</i> berdasarkan kueri pencarian yang diberikan	Berhasil
<i>Preprocessing</i>	Menampilkan hasil <i>preprocessing</i> dari <i>input</i> yang diberikan	Berhasil
Sentimen Leksikon	Menampilkan hasil pelabelan sentimen <i>InSet</i> dari <i>input</i> yang diberikan	Berhasil

Hasil pengujian *black-box* yang dilakukan pada berbagai halaman sistem menunjukkan bahwa setiap halaman berfungsi sesuai dengan harapan. Pada halaman *dataset* berhasil menampilkan informasi terkait *dataset* yang digunakan, termasuk rincian data yang telah melalui proses pembersihan (*preprocessing*) dan proses pelabelan. Pada halaman evaluasi, hasil evaluasi model seperti akurasi, *precision*, *recall*, dan metrik lainnya ditampilkan dengan benar. Halaman prediksi sentimen memungkinkan pengguna untuk melakukan *input* data baru berupa teks yang kemudian diuji menggunakan model *KNN* yang telah dikembangkan dan pengujian menunjukkan bahwa halaman ini berfungsi tanpa kesalahan. Halaman lain seperti sampel *tweet*, *preprocessing*, dan sentimen leksikon juga berfungsi dengan baik untuk menerima *input* dan mengembalikan *output*. Secara keseluruhan, setiap halaman dalam sistem berfungsi dengan baik, memenuhi tujuan pengujian, dan mendukung kelancaran analisis sentimen.

4.3 Pembahasan

Setelah dilakukannya pengujian terhadap metode *K-Nearest Neighbor (KNN)* dalam menganalisis sentimen publik terhadap pemindahan ibu kota negara, didapatkan hasil yang cukup baik. Hasil tersebut tak terlepas dari proses pelabelan menggunakan metode *Lexicon Based* serta dilakukannya proses vektorisasi kata dengan ekstraksi fitur *Term Frequency – Inverse Document Frequency (TF-IDF)* dan penyeimbangan kelas dengan salah satu metode *oversampling* yaitu *SMOTE*.

Berdasarkan hasil pengujian dan pengimplementasian yang dilakukan pada penelitian ini, penggunaan metode *K-Nearest Neighbor (KNN)* dengan menggunakan metode *gridsearch* dalam proses pencarian *hyper parameter* mendapatkan hasil yang cukup baik.

Skenario terbaik yang didapatkan pada penelitian ini adalah dengan rasio pembagian data 80:20, *weights = distance*, *metric = cosine*, dan $k = 1$ mampu menghasilkan akurasi di angka 90.18%, *recall* 90.18%, dan *precision* 90.59%. Hasil evaluasi tersebut didapatkan setelah dilakukannya penerapan metode *gridsearch* untuk mencari *hyper parameter* untuk masing-masing skenario. Proses pencarian parameter terbaik dilakukan dengan inisiasi parameter untuk menambah skenario. Dilakukan inisiasi parameter *weights* yaitu *uniform* dan *distance*. Lalu, parameter *metric* untuk mengukur jarak antara data uji baru dan setiap data yang sudah berlabel yaitu *euclidean*, *manhattan*, dan *cosine*. Terakhir yang paling penting adalah penentuan nilai k tetangga terdekat, antara lain 1, 3, 5, 7, dan 9. Selanjutnya dilakukan *cross validation* dengan nilai 10 sebagai data yang akan diuji untuk pencarian parameter terbaik. Sehingga dari proses inisiasi di atas akan dilakukan proses pencarian parameter terbaik dengan jumlah 15 kandidat dengan 10 data yang akan digunakan. Setelah dilakukannya 150 kali kemungkinan kombinasi, ditemukan parameter yang paling baik terdapat pada nilai *weights* dengan *distance*, *metric* sama dengan *cosine*, serta dengan nilai $k = 1$.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan perancangan, analisis, penelitian, serta pembahasan yang sudah dilakukan, proses penganalisisan sentimen publik terhadap pemindahan ibu kota negara menggunakan metode *Lexicon Based* untuk pelabelan dan *K-Nearest Neighbor (KNN)* untuk klasifikasi, berhasil dilakukan dengan hasil yang menunjukkan bahwa metode *KNN* dalam menganalisis sentimen memberikan kinerja yang cukup baik dalam memprediksi sentimen, baik positif, negatif, maupun netral terhadap pemindahan ibu kota negara. Penelitian ini berhasil menunjukkan bahwa metode yang digunakan dapat mencapai hasil yang cukup baik merujuk pada hasil evaluasi model di penelitian ini mencapai tingkat akurasi sebesar 90.18%, dengan nilai *recall* 90.18% dan *precision* 90.59% dengan inisiasi parameter *weights = distance*, *metric = cosine*, serta dengan nilai $k = 1$ sebagai parameter dengan hasil akurasi paling baik dibandingkan kemungkinan kombinasi yang ada setelah dilakukannya proses *gridsearch* untuk mencari parameter terbaik.

5.2 Saran

Adapun saran yang bisa digunakan sebagai landasan dalam pengembangan penelitian selanjutnya adalah sebagai berikut:

1. Penelitian selanjutnya diharapkan dapat mempertimbangkan penambahan metode seleksi fitur atau *feature selection* seperti *Chi Square*. Penggunaan seleksi fitur bisa berguna untuk memilih fitur yang relevan dan informatif, meningkatkan akurasi dan efektivitas proses klasifikasi, mengurangi beban pemrosesan, serta memotong kata yang tidak relevan.
2. Peningkatan akurasi untuk mengeksplorasi kombinasi metode atau parameter *tuning* yang lebih mendalam pada model *KNN* maupun pendekatan lain yang relevan. Selain itu, integrasi metode seperti *Random Search* dapat membantu menemukan parameter optimal, sehingga diharapkan dapat meningkatkan akurasi dan performa model dalam analisis sentimen publik terhadap pemindahan ibu kota negara.

DAFTAR PUSTAKA

- Abdillah, T. (2023). *Komparasi Metode Naive Bayes dan K-Nearest Neighbors Terhadap Analisis Sentimen Pengguna Aplikasi Zenius*.
- Afrillia, Y., Rosnita, L., Siska, D., Rigayatsyah, M., & Nurqamarina, N. (2022). Analisis Sentimen Ciutan Twitter Terkait Penerapan Permendikbudristek Nomor 30 Tahun 2021 Menggunakan TextBlob dan Support Vector Machine. *G-Tech: Jurnal Teknologi Terapan*, 6(2), 387–394. <https://doi.org/10.33379/gtech.v6i2.1778>
- Akbar, W., Astuti, E. S., & Riyadi, R. (2013). *Penerimaan dan Penggunaan Situs Jejaring Sosial Twitter di Lingkungan Mahasiswa Dengan Pendekatan Technology Acceptance Model (TAM)*. 6(1).
- Alamsyah, A. A. F., & Mulyati, S. (2023). *Implementasi Algoritme K-Nearest Neighbour dan Lexicon Based untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Gramedia Digital Pada Media Sosial Twitter* (Vol. 2, Issue 2).
- Amalia, A., Lydia, M. S., Fadilla, S. D., & Huda, M. (2018). Perbandingan Metode Klaster dan Preprocessing untuk Dokumen Berbahasa Indonesia. *Jurnal Rekayasa Elektrika*, 14(1), 35–42. <https://doi.org/10.17529/jre.v14i1.9027>
- Angel, A. C. T., Pranatawijaya, V. H., & Widiatry, W. (2024). *Analisis Sentimen dan Emosi dari Ulasan Google Maps untuk Layanan Rumah Sakit di Palangka Raya Menggunakan Machine Learning* (Vol. 4, Issue 1).
- Ayani, D. D., Pratiwi, H. S., & Muhandi, H. (2019). *Implementasi Web Scraping untuk Pengambilan Data pada Situs Marketplace*. 7(4), 257–262.
- Azhar. (2019). *Analisis Kinerja Algoritma Naive Bayes dan K-Nearest Neighbor Pada Sentimen Analisis Dengan Pendekatan Lexicon di Media Twitter*.
- Barus, S. G. (2022). *Klasifikasi Sentimen Data Tidak Seimbang Menggunakan Algoritma SMOTE dan K-Nearest Neighbor Pada Ulasan Pengguna Aplikasi Peduli Lindungi*. 162–173.
- Boeing, G., & Waddell, P. (2016). New Insights into Rental Housing Markets across the United States: Web Scraping and Analyzing Craigslist Rental Listings. *Journal of Planning Education and Research*, 37(4), 457–476. <https://doi.org/10.1177/0739456X16664789>
- Cahyana, Y., & Siregar, A. M. (2023). Analisis Sentimen Pindah Ibu Kota Negara (IKN) Baru pada Twitter Menggunakan Algoritma Naive Bayes dan Support Vector Machine (SVM). *Faktor Exacta*, 16(3), 170–181. <https://doi.org/10.30998/faktorexacta.v16i3.16703>
- Elfansyah, M. R., Rudiman, R., & Yulianto, F. (2024a). *Perbandingan Metode K-Nearest Neighbor (KNN) dan Naive Bayes Terhadap Analisis Sentimen Pada Pengguna E-Wallet Aplikasi Dana Menggunakan Fitur Ekstraksi TF-IDF*. 18(2), 139–159. <https://doi.org/https://doi.org/10.47111/JTI>
- Elfansyah, M. R., Rudiman, R., & Yulianto, F. (2024b). *Perbandingan Metode K-Nearest Neighbor (KNN) dan Naive Bayes Terhadap Analisis Sentimen Pada Pengguna E-Wallet Aplikasi Dana Menggunakan Fitur Ekstraksi TF-IDF*. 18(2), 139–159.
- Fajri, M. S., Septian, N., & Sanjaya, E. (2020). Evaluasi Implementasi Algoritma Machine Learning K-Nearest Neighbors (kNN) pada Data Spektroskopi Gamma Resolusi Rendah. *Al-Fiziya: Journal of Materials Science, Geophysics, Instrumentation and Theoretical Physics*, 3(1), 9–14. <https://doi.org/10.15408/fiziya.v3i1.16180>

- Hadi, F., & Ristawati, R. (2020). Pemindahan Ibu Kota Indonesia dan Kekuasaan Presiden dalam Perspektif Konstitusi. *Jurnal Konstitusi*, 17(3), 530–557. <https://doi.org/10.31078/jk1734>
- Ismail, A. R., & Hakim, R. B. F. (2023). Implementasi Lexicon Based Untuk Analisis Sentimen Dalam Mengetahui Trend Wisata Pantai Di DI Yogyakarta Berdasarkan Data Twitter. In *Emerging Statistics and Data Science Journal* (Vol. 1, Issue 1).
- Koto, F., & Rahmaningtyas, G. Y. (2017). InSet Lexicon: Evaluation of a Word List for Indonesian Sentiment Analysis in Microblogs. *Proceedings of the 2017 International Conference on Asian Language Processing, IALP 2017, 2018-January*, 391–394. <https://doi.org/10.1109/IALP.2017.8300625>
- Lestari, S., Mupaat, M., & Erfina, A. (2022). Analisis Sentimen Masyarakat Indonesia terhadap Pemindahan Ibu Kota Negara Indonesia pada Twitter. 8(1), 13–22.
- Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.
- Matulatuwa, F. M., Sedyono, E., & Iriani, A. (2017). *Text Mining Dengan Metode Lexicon Based untuk Sentiment Analysis Pelayanan PT. POS INDONESIA Melalui Media Sosial Twitter* (Vol. 2, Issue 3).
- Permatasari, P. A., Linawati, L., & Jasa, L. (2021). Survei Tentang Analisis Sentimen Pada Media Sosial. *Majalah Ilmiah Teknologi Elektro*, 20(2), 177–186. <https://doi.org/10.24843/mite.2021.v20i02.p01>
- Pramayasa, K., Maysanjaya, I. M. D., & Indradewi, I. G. A. A. D. (2023). Analisis Sentimen Program Mbkm Pada Media Sosial Twitter Menggunakan KNN Dan SMOTE. *SINTECH JOURNAL*, 6(2), 89–98. <https://doi.org/doi.org/10.31598>
- Pratomo, S. A., Faraby, S. al, & Purbolaksono, M. D. (2021a). Analisis Sentimen Pengaruh Kombinasi Ekstraksi Fitur TF-IDF dan Lexicon Pada Ulasan Film Menggunakan Metode KNN.
- Pratomo, S. A., Faraby, S. al, & Purbolaksono, M. D. (2021b). Analisis Sentimen Pengaruh Kombinasi Ekstraksi Fitur TF-IDF dan Lexicon Pada Ulasan Film Menggunakan Metode KNN.
- Punch, K. F. (1998). *Introduction to Social Research: Quantitative & Qualitative Approaches*.
- Putri, D. S., Sulistiyowati, N., & Voutama, A. (2023). Analisis Sentimen dan Pemodelan Ulasan Aplikasi AdaKami Menggunakan Algoritma SVM dan KNN. *Journal Sensi*, 9(2), 209–225.
- Rahayu, S., MZ, Y., Bororing, J. E., & Hadiyat, R. (2022). Implementasi Metode K-Nearest Neighbor (K-NN) untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Teknologi Finansial FLIP. *Edumatic: Jurnal Pendidikan Informatika*, 6(1), 98–106. <https://doi.org/10.29408/edumatic.v6i1.5433>
- Ramadhon, M. I. (2020). Analisis Sentimen Terhadap Pemindahan Ibu Kota Indonesia Pada Media Sosial Twitter Menggunakan Metode Algoritma K-Nearest Neighbor (K-NN).
- Raschka, S. (2018). *Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning*. <http://arxiv.org/abs/1811.12808>
- Sandi, D., Utami, E., & Kusnawi, K. (2023). Analisis Sentimen Publik Terhadap Elektabilitas Ganjar Pranowo di Tahun Politik 2024 di Twitter dengan Algoritma KNN dan Naïve Bayes. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 7(3), 1097–1108. <https://doi.org/10.30865/mib.v7i3.6298>

- Sari, R. (2020). Analisis Sentimen Pada Review Objek Wisata Dunia Fantasi menggunakan Algoritma K-Nearest Neighbor (K-NN). *Jurnal Sains Dan Manajemen*, 8(1), 10–17. www.tripadvisor.com.
- Setiawan, A., Santoso, L. W., & Adipranata, R. (2020). *Klasifikasi Artikel Berita Bahasa Indonesia Dengan Naive Bayes Classifier*.
- Sholeha, E. W., Yunita, S., Hammad, R., Hardita, V. C., & Kaharuddin, K. (2022). Analisis Sentimen Pada Agen Perjalanan Online Menggunakan Naïve Bayes dan K-Nearest Neighbor. *JTIM : Jurnal Teknologi Informasi Dan Multimedia*, 3(4), 203–208.
- Siringoringo, R. (2018). *Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan K-Nearest Neighbor* (Vol. 3, Issue 1).
- Wahid, A. A. (2020). *Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi*. <https://www.researchgate.net/publication/346397070>
- Wahyuni, R. T., Prastiyanto, D., & Suprpto, E. (2017). *Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi*.

LAMPIRAN

Lampiran 1. Sampel Data Pelabelan Manual

Teks	Validator 1	Validator 2	Validator 3
@keps_ky @rob_iqb @iwanfals Lah kan aku bilang, misal, kereta cepat ga usah dibuat, alokasikan buat daerah lain luar Jawa sana. Kan ga harus Ibu Kota pindah juga klo mau ga Jawasentris.	Negatif		
@IKNNusantaraID Ekonomi masyarakat sekitar ibukota baru akan terangkat	Positif		
@kompascom Apalagi pindahkan ibukota ya pak @prabowo ? Bukan hal yang urgent. Mhn prioritas pada 5P: Pangan, Papan (perumahan), Pendidikan, Pertahanan, Pekerjaan. Adapun Pemindahan (ibukota),	Negatif		
Pemerintah berkomitmen untuk mendorong ekosistem transportasi ramah lingkungan di Ibu Kota Nusantara (IKN), Kalimantan Timur. Dengan menggunakan energi hijau, seluruh kendaraan di IKN akan beralih ke jenis kendaraan listrik. Hal ini bertujuan untuk mewujudkan IKN sebagai smart ci https://t.co/WZAIc2WVYy	Netral		
@mas_veel Lha iya Wong IKN juga akhirnya karena Rakyat yang minta.. Demi Rakyat	Netral		
@CuitPositif Pemindahan Ibu Kota Negara jadi bukti keberanian Jokowi dalam melakukan perubahan besar untuk Indonesia. Jokowi Berhasil Publik Puas!	Positif		
Wakil Ketua MPR RI yang juga Wakil Ketua Majelis Syuro PKS Hidayat Nur Wahid menilai pernyataan Presiden Jokowi soal pemindahan ibu kota negara adalah keputusan seluruh rakyat tidak tepat. < #Jokowi #IKN #PKS https://t.co/06ypTrSzab	Negatif		
@BANGSAygSUJUD @margonohadi91 @prabowo @jokowi @Aisha5011995259 @are_inismyname @Ber_uang08 @FDonghun @genx36545403 @gtobing2903 @florielicious @Ndons_Back @XiaoMei_99 @gustavssondhela Jujungan loe pola pikirnya parah...bangun dulu ibukota sampe jadi baru jakarta dipindah..skg bingung ibukotanya dmana	Negatif		
@Andria75777 IKN blm selesai, status Jkt sebagai ibukota sdh dicabut, benar2 biadab!	Negatif		
@WidasSatyo Kami Pindah ke kota asal Ibu, terus digosipin bawa kabur uang pesangon Alm Bapak. Rumah kami ditunggu sama Adik bungsu Bapak, tp sama Nenek disewain, tiap bulan anak kandungnya dimintain duit. Bibi sampe tlp aku krn mikir aku yg narik uang sewa, pdhl mah kagak-	Netral		
pengen pindah ke tempat yang jauh entah itu luar kota atau luar negeri, buat ninggalin orang-orang yang udah nyakitin gue sama ibu gue. fuck u allðŸ–•	Netral		

Lucu banget, situ yang setuju ibu kota dipindah, eh, malah situ juga yang ogah-ogahan pindah ke sana. ðŸ™€ *ketawa sambil berguling-guling*	Negatif		
Beban Keuangan: Pemindahan ibu kota merupakan proyek besar yang membutuhkan dana yang sangat besar. Ketika kondisi ekonomi sedang sulit, proyek ini dapat membebani anggaran negara & mengalihkan anggaran dari sektor-sektor lain yg lebih mendesak, seperti pendidikan & kesehatan. https://t.co/CJ9kLjBlXu	Negatif		
@Kanseulir Lah emang ibukota mau pindah..?? Yg mau pindah cuma satu org aja.. itu orgnya sampai ngantor disana yg lain mah ogah..ðŸ™€,ðŸ™€	Negatif		
IKN Akan dil Lanjutkan di Masa Pemerintahan Presiden Prabowo Subianto. - #Indonesia #IndonesiaMaju #IKN #IbuKotaNusantara #Kalimantan #KalimantanTimur #PapuaBarat #PapuaBaratDaya https://t.co/GdWEcNRwhc	Netral		
@kompascom Kemungkinan, Joko W mau #menggertak (mengancam) Prabowo S agar tidak macam-macam & mengikuti keinginan Joko W... Apalagi hal pemindahan Ibukota RI ke Kalimantan & perlakuan terhadap Gibran R. ðŸ¥µ	Negatif		
Musrenbang Desa Padang Jaya Bahas RKPDes 2025, Fokus pada Pembangunan Berkelanjutan . . . #polri #divhumas #poldakaltim #Kaltim #IKN #Ibukotanusantara #Mahakam https://t.co/FkTXaHIAU9	Netral		
@abangbelneg Pindah ibu kota, membawa kita ke era baru yang lebih eco-chic	Positif		
Raja Jawa Ini 'Sewakan' Wilayah ke VOC, Pindah Ibu Kota, Hingga Negara Bubar https://t.co/MtBpkDhI7v	Netral		
@realfedinuril Bung Karno ingin pindah ibukota atau engga udah ga relevan. Pindah ibukota itu keharusan. Faktanya beban Jakarta sudah terlalu berat.	Positif		
@jokowi Pak moelyo-no tdk betah di IKN ? Kemarin statemen nya mau tinggal di IKN selama 40 hari,isuk delai sore tempe	Negatif		
@BebySoSweet pemerintah terlalu gegabah, grasa grusu, tidak pruden, telah mengesahkan Undang-Undang Nomor 2 Tahun 2024 tentang Daerah Khusus Jakarta sementara IKN Nusantara belum berfungsi Indonesia kehilangan status ibukota Negara ('the nationâ€™s capitallessâ€™) https://t.co/itE9Z3UbM8	Negatif		
Giliran gatot katanya rakyat yg minta, rakyat minta ga macet, ga mahal pajak, bpjs ga ribet dan berkualitas pelayanannya,	Negatif		

akses jalan bagus, akses listrik rata, akses pendidikan mudah, akses internet mudah, GA minta ibu kota baru!!!!!!!!!!!!			
@Fahrihamzah Daerah khusus nya udh disah kan,,knp ibu kotanya ga sklian disahkan? Artinya pemindahan ibukota ini dipaksakan n sy pribadi curiga ada tekanan dr phak luar om..pasal 63 klo daerah khusus ya khusus aj donk knp ga dialihkan smntara ke daerah istimewa? Artinya Mulyono cuci tgn donk	Negatif		
@realfedinuril @jokowi opini pribadi gue sbg warga sipil, yaaa IKN cuma preyek ambisius jokowi aja buat nyari validasi	Negatif		
@tempodotco Bukan jaminan pindah Ibu kota pembangunan akan merata	Negatif		
Keputusan Presiden (Keppres) pemindahan ibu kota keÂ Ibu Kota NusantaraÂ (IKN) belum ditandatangani hingga kini. Apa alasannya? https://t.co/Q0aLG167kA	Netral		
Kartu Truf utk Menyandera ? Jokowi belum tandatangani KEPPRES pemindahan ibukota ke IKN. SeringÂ²lah Puja puji Paduka ! https://t.co/AQSKJcGyUI	Negatif		
wkwkwkwk akhirnya nemu orang yg make uu ini bahas ikn anjaaay	Netral		
@asumsico Boy, jgn ngomong doang. Buktiin rakyat mana yang memutuskan ibukota pindah??!	Negatif		
gw ke org Delivery "Projek IKN yg kemarin, udh jadi dikirim pak?" "Udh kmrn Kamis" "Lah ngapain dikirim? Kan ibu kota ga jd pindah ke sana" "Ya biarin, plg juga mangkrak" hahahahaha	Negatif		
@mazzini_gsp tapi jakarta mau tenggelam bang, mau ada wacana atau gak ada dari jaman soekarno yg jelas ibu kota harus pindah dan udah setengah jalan	Positif		
@MardaniAliSera @PKSejahtera Kemarin menentang pemindahan ibukota....hari ini adlh pengagum ikn....tercerahkan ato plin-plan...??	Netral		
@ferrykoto @aniesbaswedan @PDI Perjuangan Sebentar lagi? Mau taruhan sama saya mas berapa lama lagi ibu kota pindah? Pin aja nih comment saya menurut sumber busa 10-20 tahun ke depan. Di kira pindah administrasi ibu kota ga lama, seenaknya buat narasi kaya orang ga berotak wk!	Negatif		
@mastbagust @Hilmi28 Model pindah ibukota knp tdk dibarengi TAP MPR, mn Hak & kewajiban mandataris & mn hak kewajiban @mprgoid @DPR_RI	Negatif		

program" Nasional wajib final nya penatanganan ada di DPR, kemen, BUMN, POLRI TNI & badan lainnya dgn pembiayaan besar skala nasional wajib di TDO olh DPR			
Pemerintah berkomitmen untuk mendorong ekosistem transportasi ramah lingkungan di Ibu Kota Nusantara (IKN), Kalimantan Timur. #IbuKotaNegaraNusantara #IKN #IKNNusantara #IndonesiaEmas2045 #SmartCity #KotaModern #IndonesiaMaju #SmartDefenseSystem #KerjaNyataJokowi https://t.co/ILywI9SQ0f	Netral		
@AgikMauripta Pemindahan ibu kota mencerminkan visi Indonesia sebagai negara yang berkeadilan	Positif		
@garudadidada10 Pemindahan ibu kota mengarahkan pembangunan ke daerah yang lebih potensial	Positif		
@zola_papazola2 @msaid_didu Maaf bang mau nanya, menurut ABG tahun berapa ibukota layak pindah dan siapa presidennya, menurut ABG presiden pada tersebut sanggupkah mengambil keputusan, dan menjelang ibukota pindah sampai tahun tersebut, boleh saya tahu kondisi jakarta, karena bangun ibukota butuh 40 tahun	Positif		
@kompascom Sepertinya akan gagal total pemindahan ibu kota harinya dia mau cuci mulut https://t.co/UmSHFxDyHO	Negatif		
Simak selengkapnya dalam 'Salam Bukit Raya' di channel YouTube Pemerintah Desa Bukit Raya ya, Sob! #BukitRaya #Sepaku #PenajamPaserUtara #KalimantanTimur #IbuKotaNusantara #PTSL	Netral		
Para Pendukung Ibukota Nusantara terhimpun dalam Forsa IKN melaksanakan secara Bersama acara tatap muka yang dihadiri oleh lebih dari 300 orang peserta yang datang dari berbagai kota-kota. https://t.co/aGEBbvaN4y Speed Tawangmangu READY BO Mossad Lebanon Labubu Garut Sedimen https://t.co/Iwm8nlzktq	Netral		
@aldapstr "pasti sudah dikaji dulu" IKN noh tanpa amdal	Negatif		

@Putri_ajha kalian sadar ga sih keknya pemerintah bakal tau kalo jawa makin lama akan tenggelam atau mengecil seperti rumor tanggul dan gempe megatrast makanya ibu kota pindah IKN soalnya pulau indo ini kecil klo diliat dari benua/negara lain	Positif		
Menteri Koordinator Kemaritiman dan Investasi Luhut Binsar Pandjaitan menyampaikan bocoran terkait rencana pembangunan Ibu Kota Nusantara (IKN) di era pemerintahan Prabowo Subianto. #luhut #prabowo #jokowi #pembangunan #eraprabowo #hubungan #ibukotanusantara #ikn https://t.co/MiFAAmIldO	Netral		
@Tan_Mar3M Beliau baru sadar pindah ibu kota tidak semudah pindah rumah	Negatif		
@_MbakSri_ Setelah ibu kota negara pindah kaliamantan timur habislah politik islam pulau jawa	Negatif		
Gara2.mau pindah ibu kota bulan juni gk jadi. Jadinya otak pindah kelutut. Cebok kiri kanan Pasang selimut setebal mungkin. Seperti ferry kotok, Yg berbicara slalu tidak dgn pakta tpi dgn kebencian. Ujung2nya..ee kebo hihhi	Negatif		
@ardisatriawan melihat jakarta sangat panas, banyak kendaraan, trotoar, KRL yang full aku rakyat yg menyepakati Ibukota pindah ðŸ˜Œ	Positif		
Dengan menggunakan energi hijau, seluruh kendaraan di IKN akan beralih ke jenis kendaraan listrik. #IbuKotaNegaraNusantara #IKN #IKNNusantara #IndonesiaEmas2045 #SmartCity #KotaModern #IndonesiaMaju #SmartDefenseSystem #KerjaNyataJokowi #JokowiMerakyat #JokowiBerdedikasi https://t.co/W8JuWjuISo	Netral		
Ingat, Dilarang Merokok Saat Berkunjung ke IKN < #IKN #IbuKotaNusantara #OIKN https://t.co/qHo9Kl2TJ4	Netral		
@Hasbil_Lbs @budjaya wkwwkwkw padahal dikasih artikel nya pun, based on cases yg sama terkait dgn pemindahan ibukota. ga cuman asumsi belaka. susah emang kalau udah dapet "kue". jilat teros lahhh, gasss ~	Negatif		
Rencana Pertanian Modern di IKN, Kegagalan Food Estate Belum Cukup? https://t.co/1Otjn2qomG	Negatif		
@tempodotco Sebenarnya yg mau buru-2 ibukota pindah itu siapa sih...? Ko jadi kebalik-2.	Netral		
@zy_zy_lestary Penulis Inggris John Heywood pernah menulis dalam karyanya â€œRoma non fuit condita uno dieâ€œ. Roma tidak dibangun dalam satu hari	Positif		

Ungkapan ini untuk mengingatkan bahwa hal-hal besar membutuhkan waktu dan usaha yang konsisten untuk tercapai Tuntasnya proses pemindahan ibukota dan			
@Zaoffar @Kanseulir Ya memqng harus pindah ibukota.... Mosok cm jawa aja dibangun? Indonesia cuma Jawa aja? Aya2 wae....	Positif		
Akar rumput haters Jokowi dr gerombolan kalah pilpres, sdh tertanam doktrin IKN mangkrakðŸ- Aku sbg pendengar ocehan dia cukup senyam senyum sj. Krn aku slh satu voters 02, pendukung 02. Hrpn rakyat 02 semua program Jokowi yg blm tuntas selesai o) 02 https://t.co/Z99VXyYKjw	Positif		
Perdana, Kemenag Buka Lowongan CPNS untuk Lulusan Maâ€™™had Aly: Ada Kuota Khusus Putra/Putri Kalimantan https://t.co/O47sdXFbRV #ikn #ibukotanusantara #iknnusantara #indonesiaemas2045 #dukungpembangunanikn #iknmajukanindonesia #iknkotakelasdunia #investorikn https://t.co/LvUkWbwL7a	Positif		
@abu_waras Dimana ibukota negara indonesia? Yo Ndak Tahu Saya Kok Tanyakan Saya Jakarta bukan, IKN belum di teken	Negatif		
Telan Anggaran Rp27,6 Triliun, Pembangunan IKN Tahap II Mencakup 34 Paket Pekerjaan https://t.co/p6BtqpJ7RI #ikn #ibukotanusantara #iknnusantara #indonesiaemas2045 #dukungpembangunanikn #iknmajukanindonesia #iknkotakelasdunia #investorikn #ibukotanegara #indonesiamaju https://t.co/5piU7Adhc0	Netral		
Wow teori pemindahan ibukota yang menarik	Positif		

@bokulos_ @tempodotco "Kita semua tahu, ANGK0T88. IKN bukan untuk rakyat, tapi untuk kepentingan politik semata. #Preman #MUNTOT"	Negatif		
@UmarSyadatHsb_ Emang mulyono ini mulutnya mencla-mencele gak jelas banget. Dia yg paling ngoto bikin ikn, eh dia pula yg ngomong pindah ibukota gak gampang.	Negatif		
@cloudrops disenggol gapura kabupaten bisa langsung pindah ke ibu kota lu	Netral		
@democrazymedia Rakyat tidak pernah berbicara soal pemindahan ibukota.	Negatif		
@DedynurPalakka 2 periode mencetak banyak sejarah bener ini Mecahin rekor utang negara Sejarah baru dpr kompak sampai undang2 bisa dibuat hitungan hari aja Sejarah baru ibukota pindah, Eh udah pindah blm ya?? Sejarah baru, matinya oposisi Sejarah baru korrupt 417t denda 5k Akui aja SBY lbh jos	Negatif		
@tempodotco Silahkan saja .. mungkin mau menjauh dari demo. Itu bukan berarti Ibu Kota Negara Republik Indonesia sudah pindah kesana.. masih tetap Jakarta. Apasih maksud singkatan IKN? Ibu Kota Nusantara? Negara kita bernama Indonesia bukan Nusantara. cc: @alisyarief @PreciosaKanti	Negatif		
Penandatanganan Keppres Pemindahan Ibu Kota, Presiden: IKN Nusantara Harus Semuanya Sudah Siap https://t.co/XYs8qazhGQ via @BeritaIn https://t.co/4rThnSOGmv	Netral		
temen gue ditanya keuntungan pembangunan ikn untuk rakyat apa, dia jawab 'jakarta kan mau tenggelam, jd ibu kotanya di pindahkan lah' TRS GUE KEK HAH? sambil melengos ajah. eh dia lanjut ngomong 'lagian ikn udh direncanakan dr zaman soekarno' ðŸŹŹðŸŹŹ-	Negatif		
Alhamdulillah uang pajak saya turut berkontribusi 1 buah tiang lampu di taman IKN Gimana? Sdh puas dgn rencana IKN dr pmrintah? Agar lebih cepat realisasi IKN sampai final, semangat bayar pajak semua!! #ikn #ibukotanusantara #kaltim #ppu https://t.co/pDmyxGDSPS	Negatif		
@Leonita_Lestari UU Nomor 2 Tahun 2024 Jakarta MASIH menyandang status Ibu Kota Negara sampai adanya Keppres tentang pemindahan ibu kota negara ke Nusantara.	Negatif		

Setelah dipikir2 Presiden berikutnya, Keppres pembatalan mas. Esemka aja 10 tahun gak jelas, padahal cuma mobil roda 4			
Klaim Jokowi: Rakyat yang Ingin Ibu Kota Pindah ke IKN https://t.co/MGIgCnSm4Y	Netral		
Terserah ya mau ngomong apa.. tapi gua setuju2 aja dengan pemindahan ibu kota dengan salah 1 alasan mengurai kemacetan jkt. Apalagi proyek udah jalan gini.. tolol amat kalo gajadi. Rugi dong.	Positif		
@tempodotco Sampah..got ikn aja nolak	Negatif		
Jokowi Frustasi soal IKN? Jakarta Masih Ibu Kota RI sampai Ada Keppres Pemindahan ke IKN - https://t.co/6XQZ0kpibo	Netral		
Bhabinkamtibmas Kelurahan Handil Bakti Bersama Pelajar SD 020 Lakukan Penghijauan Lingkungan . . . #polri #divhumas #poldakaltim #Kaltim #IKN #Ibukotanusantara #Mahakam https://t.co/aWQi95JqMv	Netral		
Delonix Nusantara Groundbreaking di IKN, Investor Asing yang Berinvestasi Secara Langsung di Nusantara https://t.co/CFQnB5MOL2 #ikn #ibukotanusantara #iknnusantara #indonesiaemas2045 #dukungpembangunanikn #iknmajikanindonesia #iknkotakelasdunia #investorikn https://t.co/jr0fVbl7iN	Netral		
INFO TEMPO-Perubahan Undang-Undang IKN diyakini akan memberikan manfaat serta penguatan untuk menghadirkan sebuah ibu kota negara yang baru, yang menjadi salah satu milestone langkah negeri ini untuk menjawab tantangan masa depan serta menggapai cita-cita dan visi Indonesia Emas https://t.co/y145TV4Wda	Positif		
@abangbelneg Pemindahan ibu kota, bukti bahwa kita serius dalam mengelola sumber daya alam.	Positif		
Pengguna transportasi umum di Jakarta, pasti tidak asing lagi dengan Stasiun Jatinegara. Stasiun Jatinegara sudah menjadi cagar budaya sejak tahun 2005, memiliki banyak kisah penting dalam perjalanan sejarah Indonesia, salah satunya adalah sebagai saksi bisu pemindahan Ibu Kota https://t.co/KhCEYwiHtW	Netral		

@pakmul63 Mulyono tuh memang konyol banget, sok-sokan pindah ibukota, eh dia sendiri mau pensiun di Karanganyar, mencla-mencle.	Negatif		
PUPR: Progres Tahap I Pembangunan IKN Capai 92 Persen https://t.co/cuJtdMW793 #ikn #ibukotanusantara #iknnusantara #indonesiaemas2045 #dukungpembangunanikn #iknmajukanindonesia #iknkotakelasdunia #investorikn #ibukotanegara #indonesiamaju #pembangunanikn https://t.co/kQg7um9fC7	Netral		
@IKNnusantaraID Kami warga Bengkulu mendukung pemindahan Ibu Kota negara	Positif		
Pindah ibu kota ke lokasi bukan lembah, bebas banjir dan tanah jerlus “ Puad #malaysiagazette https://t.co/XTGKVhDHQO	Positif		
Pimpin Apel Pagi, Wakapolda Kaltim Ucapkan Terima Kasih Atas Dedikasi Personel Serta Ingatkan Netralitas Selama Pilkada . . . #polri #divhumas #poldakaltim #Kaltim #IKN #Ibukotanusantara #Mahakam https://t.co/ekALLrCbeO	Netral		
Presiden Jokowi telah memberikan kontribusi besar berbagai proyek infrastruktur di seluruh Indonesia. Proyek infrastruktur ini fokus pada jalan tol, pelabuhan, bandara, dan jaringan kereta api, serta pembangunan IKN. *** #TerimakasihPakJokowi 1 Oktober Rafi Eliano Reijnders Mbak https://t.co/coRzZyUMsy	Positif		
@meraaahputiih Pindah ibu kota, bikin kita semakin siap untuk meraih prestasi dunia.	Positif		

<p>Kementerian ATR/BPN Klaim Selesaikan 13 Paket Pengadaan Tanah di Kawasan Inti IKN</p> <p>https://t.co/CuvJf1wmu4</p> <p>#ikn #ibukotanusantara #iknnusantara #indonesiaemas2045 #dukungpembangunanikn #iknmajukanindonesia #iknkotakelasdunia #investorikn #ibukotanegara #indonesiamaju https://t.co/nMv1Ga7Vzu</p>	Netral		
<p>@realfedinuril @jokowi Kalo dilihat sesuai konteks saat itu, kayanya pengukuhan Jakarta sebagai ibukota setelah sebelumnya pindah2 karena situasi keamanan.</p> <p>Beda konteks utk IKN ini. Tapi memang perencanaannya harus super duper matang. IKN ini terlalu buru2.</p>	Negatif		
<p>Di Istana Negara IKN Jokowi Tegaskan Ingin Pulang ke Solo</p> <p>https://t.co/AaoE5Eiraj</p> <p>#ikn #ibukotanusantara #iknnusantara #indonesiaemas2045 #dukungpembangunanikn #iknmajukanindonesia #iknkotakelasdunia #investorikn #ibukotanegara #indonesiamaju #pembangunanikn https://t.co/13yMHBVXjj</p>	Netral		
<p>@tempodotco @Android_AK_47 awalnya dia ingin membuktikan pada dunia..bhwa si plonga plongo ini hebat bisa mewujudkan ibukota baru..endingnya justru IKN ini justru melegitimasi keplonga plongo annya..</p>	Negatif		
<p>@TirtoID Banyak UU kalo isinya ngawur ya ngapain. UU Ciptaker, UU ITE, UU IKN, semuanya UU sampah yg ga guna buat rakyat.</p>	Negatif		

@mas_veel Bang apa benar ikn itu proyek rakyat atau proyek jokowi?	Netral		
@starfess yg dipakai apa? duit pajak. duit negara, ngutang, trs pindah ibu kota biar apa ya tau sendiri..	Negatif		
@nieluniverse @falfallin @Marchfoward Kalimantan Just 20 million people, kalau je ada demo tentang pemerintahan, akan sulit mengakses masyarakat untuk bersuara di depan istana, intinya banyak dari kami kecewa pemindahan Ibukota di pindahkan sepihak, lalu menjadikan IKN dari tanah kosong menjadi ibu kota tanpa rakyat.	Negatif		
Jokowi: Pindah Ibu Kota Bukan Keputusan Presiden Saja, tapi Seluruh Rakyat Keluarga Mulyono, fufufafa dan clara ikut juga gak pak Jokowi https://t.co/zh0HCO77xG	Netral		
Banyak yang Belum Tahu! Pemindahan Ibu Kota ke IKN Ternyata Bukan Keinginan Jokowi Melainkan Ide dari Sosok Legendaris Ini https://t.co/ICKQ9DmunY	Netral		
@BosPurwa Mirip proyek kota baru di Lampung nih... Proyek pemindahan ibukota provinsi Lampung yg gagal dan akhirnya terbengkalai...	Negatif		
@meraaahputiih Pemindahan ibu kota memberikan peluang besar untuk perkembangan nasional.	Positif		
@IndonesiaJadi Penajam Paser Utara bakal jadi penyuplai pangan utama buat IKN Nusantara	Positif		