ANALISIS SENTIMEN PENGGUNA TWITTER TERHADAP IBU KOTA NUSANTARA (IKN) SETELAH DEBAT CALON WAKIL PRESIDEN TANGGAL 22 DESEMBER 2023 MENGGUNAKAN METODE NAIVE BAYES DAN K-MEANS

LAPORAN AKHIR TUGAS BESAR

Oleh:

David Ephraim | 105221002

Arya Ashari | 105221013

Christo Zwingly Alexander | 105221019

FAKULTAS SAINS DAN KOMPUTER

PROGRAM STUDI ILMU KOMPUTER

UNIVERSITAS PERTAMINA

2024

DAFTAR ISI

BAB I. PENDAHULUAN

1.1. Latar Belakang

Pemindahan Ibu Kota Negara (IKN) dari Jakarta ke Kalimantan Timur merupakan salah satu kebijakan pemerintah Indonesia yang paling kontroversial dalam beberapa tahun terakhir. Kebijakan ini menimbulkan berbagai opini masyarakat yang berbeda, mulai dari yang mendukung hingga yang menentang.

Media sosial, khususnya Twitter, menjadi salah satu platform yang digunakan masyarakat untuk menyampaikan opininya mengenai IKN. Berdasarkan hasil penelitian sebelumnya, diketahui bahwa sentimen masyarakat terhadap IKN di Twitter didominasi oleh sentimen negatif. Hal ini menunjukkan bahwa masyarakat masih memiliki banyak keraguan terhadap IKN, terutama terkait dengan biaya pembangunan, dampak lingkungan, dan keadilan bagi masyarakat di Kalimantan Timur.

Debat calon wakil presiden yang diselenggarakan pada tanggal 22 Desember 2023 menjadi salah satu momentum penting untuk mengetahui pandangan masyarakat mengenai IKN. Dalam debat tersebut, kedua pasangan calon wakil presiden menyampaikan pandangannya masing-masing mengenai IKN.

Penelitian ini bertujuan untuk menganalisis sentimen pengguna Twitter terhadap IKN setelah debat calon wakil presiden. Penelitian ini diharapkan dapat memberikan gambaran mengenai pandangan masyarakat terhadap IKN setelah mendengar pandangan dari kedua pasangan calon wakil presiden.

1.2. Tujuan

Tujuan penelitian ini adalah untuk:

- Mengetahui sentimen pengguna Twitter/X terhadap IKN setelah debat calon wakil presiden.
- Mengidentifikasi faktor-faktor yang mempengaruhi sentimen pengguna Twitter terhadap IKN setelah debat calon wakil presiden.

1.3. Manfaat

Manfaat penelitian ini adalah untuk:

- Memberikan gambaran mengenai pandangan masyarakat terhadap IKN setelah mendengar pandangan dari kedua pasangan calon wakil presiden.
- Memberikan informasi kepada pemerintah mengenai sentimen masyarakat terhadap IKN.

- Menjadi referensi bagi penelitian-penelitian selanjutnya mengenai sentimen masyarakat terhadap IKN.

BAB II. METODE

2.1. Analisis Sentimen

Analisis sentimen merupakan cara untuk mengelompokan opini maupun respon publik terhadap suatu masalah atau topik. Pengelompokan dari opini publik dibagi menjadi tiga yaitu, positif, negatif, dan netral. Adapun tujuan dari melakukan analisis sentimen yaitu untuk mengetahui opini publik yang setelah dianalisis dapat dihasilkan rekomendasi untuk melakukan perubahan, perbaikan atau insight lainnya (Majid et al., 2023).

2.2. Naïve Bayes

Naïve Bayes adalah algoritma klasifikasi probabilistik yang menggunakan teori probabilitas Bayes. Metode ini bekerja dengan menghitung probabilitas suatu data termasuk ke dalam kategori tertentu berdasarkan fitur-fitur yang ada dalam data tersebut. Salah satu algoritma yang biasa digunakan untuk melakukan klasifikasi. Adapun klasifikasi yang dimaksud yaitu klasifikasi yang digunakan untuk melakukan analisis sentimen. Algoritma ini digunakan kaena mampu memberi penghitungan cepat dengan tingkat akurasi yang cukup baik (Affandi & Sugiharti, 2023).

2.3. K-Means

K-Means adalah algoritma unsupervised learning dalam machine learning yang digunakan untuk mengelompokkan data tidak berlabel ke dalam kluster yang berbeda. Algoritma K-Means akan menggabungkan data-data yang memiliki tingkat kesamaan dari karakteristik yang tinggi pada satu kelompok. Perbedaan pada data akan dimuat ke dalam kelompok baru (Nainggolan et al., 2022).

BAB III. HASIL DAN PEMBAHASAN

3.1. Teknik Pengambilan Data

Pengumpulan data dilakukan pada rentang waktu antara 22 Desember 2023 hingga 21 Januari 2024 dengan menggunakan metode *web scraping* (scrapping) menggunakan kata kunci "IKN" dan *hashtag* "#IKN". Hasil dari pencarian ini menunjukkan bahwa terdapat 811 tweet yang menggunakan kata kunci "IKN" dan 307 tweet yang mencakup *hashtag* "#IKN".

Dengan menggabungkan kedua kelompok data tersebut, diperoleh total 1118 tweet. Namun, setelah melakukan proses penghapusan *duplicate* atau duplikasi, ditemukan bahwa ada 1059 tweet yang unik dan relevan dalam analisis data ini.

Proses pengumpulan data ini memiliki implikasi signifikan dalam memahami tren atau perbincangan terkait dengan topik "IKN" di platform Twitter selama periode waktu yang telah ditentukan. Dengan memiliki dataset yang telah disaring dari duplikasi, analisis lebih lanjut dapat dilakukan untuk mengeksplorasi pola atau sentimen yang mungkin terkandung dalam tweet-tweet tersebut.

Penting untuk diingat bahwa data yang diperoleh melalui *web scraping* memiliki beberapa batasan, termasuk potensialnya adanya noise atau informasi yang tidak relevan. Oleh karena itu, interpretasi hasil analisis perlu dilakukan dengan hati-hati, dan sumber data harus diperhatikan dalam konteks kredibilitas dan keandalannya.

3.2. Teknik Pembersihan Data

Terdapat beberapa teknik pembersihan data yang dilakukan pada dataset. Teknik ini umumnya digunakan ketika kita bekerja dengan teks dari platform Twitter. Berikut adalah penjelasan untuk setiap teknik pembersihan:

1. Menghapus Mention (@[A-Za-z0-9_]+):

   Pada langkah ini, menggunakan regular expression (re), semua *mention* atau username yang diawali dengan "@" dihapus dari teks. Ini bertujuan untuk menghilangkan informasi pengguna Twitter yang disebutkan dalam tweet.

2. Menghapus Hashtag (#\w+):

   Dengan menggunakan regular expression, semua *hashtag* yang dimulai dengan tanda "#" dihapus dari teks. Ini dilakukan untuk menghilangkan tagar dan memusatkan analisis pada kata-kata kunci tanpa karakter khusus.

3. Menghapus Retweet Indicator (RT[\s]+):

Dalam langkah ini, menggunakan regular expression, menghapus indikator *Retweet* ("RT") dan spasi yang mungkin mengikuti indikator tersebut. Hal ini dilakukan agar fokus analisis lebih pada teks sebenarnya tanpa indikator retweet.

4. Menghapus URL (https?://\S+):

   Dengan menggunakan regular expression, semua URL dihapus dari teks. Ini dilakukan untuk menghilangkan tautan atau *link* yang dapat mengganggu analisis dan memusatkan perhatian pada konten teks.

5. Menghapus Karakter Spesial dan Simbol ([^A-Za-z0-9()]):

   Dalam langkah ini, menggunakan regular expression, karakter spesial dan simbol selain huruf, angka, dan tanda kurung dihapus dari teks. Hal ini dilakukan untuk menjaga keberagaman karakter dan membuang simbol yang mungkin tidak relevan dalam analisis.

6. Menghapus Spasi Berlebih (\s+):

   Dalam dua langkah ini, spasi berlebih dihilangkan dan diubah menjadi satu spasi tunggal. Hal ini untuk menjaga konsistensi dan mempermudah analisis data.

7. Menghapus Tanda Kurung (\(|\)):

   Dengan menggunakan regular expression, tanda kurung (baik kurung buka "(" maupun kurung tutup ")") dihapus dari teks. Ini dilakukan untuk menghilangkan tanda kurung yang mungkin tidak relevan dalam konteks analisis.

8. Menghapus Tanda Baca ([.,]):

   Dengan menggunakan regular expression, tanda baca seperti titik dan koma dihapus dari teks. Ini membantu membersihkan teks dari elemen-elemen yang mungkin mengganggu analisis.

   Setelah semua langkah pembersihan dilakukan, teks yang tersisa diubah menjadi huruf kecil semua menggunakan metode `str.lower()`. Hal ini dilakukan untuk memastikan konsistensi dalam analisis teks, mengingat pengejaan huruf besar dan kecil dianggap setara dalam analisis teks.

3.3. Preprocessing

Teknik preprocessing yang dilakukan pada kode di atas mencakup beberapa langkah penting untuk membersihkan dan mengolah data teks, termasuk normalisasi, penghapusan stopwords, dan tokenisasi dengan stemming. Berikut adalah penjelasan untuk setiap teknik preprocessing yang terdapat dalam kode tersebut:

1. Normalisasi:

Normalisasi dilakukan dengan menggantikan beberapa bentuk kata-kata informal atau singkatan dengan bentuk formal atau lengkapnya. Contoh normalisasi yang dilakukan antara lain mengganti "yg" menjadi "yang", "tdk" menjadi "tidak", "utk" menjadi "untuk", dan sebagainya. Normalisasi bertujuan untuk membuat data teks lebih konsisten dan mudah dipahami.

2. Penghapusan Stopwords:

   Penghapusan stopwords dilakukan dengan menggunakan kamus stopwords dari Sastrawi dan menambahkan beberapa stopwords tambahan yang dianggap kurang informatif. Stopwords adalah kata-kata umum yang sering muncul dalam teks namun tidak memberikan informasi yang signifikan. Penghapusan stopwords bertujuan untuk meningkatkan akurasi analisis teks dengan menghilangkan kata-kata yang tidak relevan

3. Tokenisasi dan Stemming:

   Tokenisasi dilakukan dengan memecah teks menjadi token (kata-kata) secara individual. Stemming dilakukan untuk mengubah kata-kata menjadi bentuk dasarnya dengan menghapus akhiran atau imbuhan. Proses tokenisasi dan stemming membantu mengurangi variasi kata dalam teks, sehingga dapat meningkatkan konsistensi dan efisiensi analisis teks.

4. Penyimpanan Data:

   Data hasil preprocessing disimpan dalam format CSV untuk digunakan dalam tahap analisis selanjutnya. Penyimpanan data dilakukan agar proses preprocessing tidak perlu diulang setiap kali melakukan analisis, dan data yang telah bersih dapat dengan mudah diakses.

Teknik preprocessing ini membantu mempersiapkan data teks sehingga lebih siap digunakan untuk analisis sentimen atau tugas pemrosesan bahasa alami lainnya. Data yang telah melalui proses preprocessing tersebut akan lebih terstruktur, konsisten, dan relevan untuk proses analisis selanjutnya.

3.4. Analisis

Kami melakukan plotting dengan wordcloud dalam rangka melihat kata yang sering digunakan dalam perbincangan pada media sosial mengenai IKN. Pada Gambar 3.1 dapat dilihat bahwa kata-kata yang sering muncul atau yang sering digunakan untuk berbincang mengenai proyek IKN yaitu “ikn”, “bangun”, “nusantara”, “jadi”, “lanjut” dsb. Adapun alasan semua kata yang terdapat pada wordcloud tidak ada yang kapital dikarenakan sebelum melakukan plotting, kami melakukan *casting* menjadi *lowercase* terhadap semua kata agar dapat mempermudah pengolahan data kedepannya.

Gambar 3. 1 Cloudword

Setelah melakukan plotting menggunakan wordcloud terhadap data setelah preprocessing, kami melakukan plotting dengan cloudword terhadap tiga jenis sentimen, yaitu sentimen positif, negatif, dan netral. Hal ini bertujuan untuk melihat kata-kata apa saja yang sering digunakan dan masuk ke dalam kelompok sentimen. Pada Gambar 3.2 merupakan hasil wordcloud untuk sentimen positif masyarakat terhadap proyek IKN.

Gambar 3. 2 Cloudword klasifikasi positif

Selanjutnya, kami melakukan plotting yang kedua untuk mengetahui kata-kata yang sering digunakan dan masuk ke dalam klasifikasi dari sentimen negatif terhadap proyek IKN. Hasil dapat dilihat pada Gambar 3.3.

Gambar 3. 3. Cloudword klasifikasi negatif

Terakhir, kami plot wordcloud untuk sentimen netral terhadap proyek IKN yang dapat dilihat pada Gambar 3.4.

Gambar 3. 4. Cloudword klasifikasi netral

Berikut ini adalah daftar 10 kalimat dengan masing-masing sentimen yang ada.

Tabel 3. 1. Tweet positif dengan klasifikasi textblob

10 Tweet dengan klasifikasi positif
bagus sekali tag ke ahy demokrat reuni sama candi hambalang ikn odong-odongan
prabowo klaim ikn puas masyarakat kalimantan rakyat sangat antusias
345 investor minat garap proyek ikn antara china arab saudi oikn catat 345 nyata minat investasi investor dan negeri ingin ikut gabung garap ikn nusantara nana mirdad bahrain bal bank mandiri yessica tamara

sangat logis orang saat tutup logika pikir prororo cuma wayang master pakdhe bukti program kerja 02 lanjut program pakdhe nah coba bayang bakal nego sama investor ikn siapa prororo lanjut pakdhe sekali ngorbitin samsul
bp nuarta patung senior punya staff arsitek lisensi masa nuntut kuliah arsitek dulu baru bangun ikn lucu nama tadao ando kuliah arsitek puluh staffnya jago jago lha masa bos suruh kuliah aneh
otorita ibu kota nusantara oikn catat 345 letter of interest loi alias nyata minat investasi investor dan negeri ingin ikut gabung garap ikn nusantara nana mirdad bahrain bal bank mandiri yessica tamara
pak minta buat ajar soal masyarakat adat wilayah ikn bukan buat nyerang buat siap kalo tanya masalah ini
buka jati diri manusia masing masing jadi pilih no 1 tuju ikn hilir sda indonesia kita tahu kubu pilih no 2 karena sifat orde baru terap korupsi kecil kecil amp yang penting rakyat kenyang no3 cerdas
iya benar jga awal tuju tidak jawasentris jdi simbol tpi kalau lihat ada rakyat indo ini buat makan susah lho ikn terlalu buru visi amin itu kita buat masyarakat mandiri dlu cari uang sendiri sejahtera bru ikn
nah bahas perata ka ikn sudah jalan mas gibran sendiri sudah nyata ini simbol perata bangun indonesia arti tidak langsung siapa presiden tetep laku perata betul

Dari tweet yang masuk klasifikasi positif sebagian besar dapat dikatakan positif. Dari 10 tweet diatas 3 diantaranya masih terdengar seperti tweet negatif.

Tabel 3. 2. Tweet negatif dengan klasifikasi textblob

10 Tweet dengan klasifikasi negatif
yah dayaknya juga dukung ikn mau maju sendiri daerah mikirin daerah laen
aku sempat diskusi dikit aku kalah soal gabisa jawab ikn yang tuju anies terang terang tolak ikn alas ikn perlu bagus akan maju kalimantan timur aku memang kerja salah satu proyek ikn makan pro akan sejarah ulang reklamasi proyek bohir rezim gagal anies jadi gubernur gagal reklamasi muncul ikn proyek bohir taruh besar anies jadi presiden

kalau aku orang tua sih masalah ikn soal tinggal kalimantan orang tua gasuka anies kata anies tolak ikn aneh2 sih aku rasa cukup penting
saat tempat suatu provinsi papua ikn lebih dekat jawa homebase dan lihat ada alam baru seru jadi ke homebase gakpapa
orang butuh kerja uang makan bukan ikn mau ini tolol
anies ganjar sama sama kualitas sih bkin beda ikn kak tinggal di sulawesi mungkin kk kurang ngerti tpi timpang sosial bangun jauh gila dana anggar all in ikn yang dri tanah saya lebih baik perata
ubah ikn jadi hutan rimba ubah jalan tol jadi sawah ubah umk kita dari 4 500 000 - jadi 1 500 000 - kita ubah presiden kita dari orang pribumi jadi orang asing yaman ngeri kali ustadz kita ini bah
hahahha liat banget kau tolol sekarang investor asing udh sedia invest ikn berapa nol arti kalau ikn lanjut bakal pake apbn trus otak kau bilang dana bagi besar negara lain dukung 02 bodoh kayak kau gin kah
jauh si banyak masalah ikn tapi kemarin sempat liat gagas anies pas desak anies jadi sama ngertii akhir buka pikir yang pindah juga tetep bingungg wkwk
yang pikirin soal ikn org2 kalimantan kata yang tuju perata kata kalimantan maju seperti jakarta pdhl sadar egois banget nget nget bener2 buang anggar buat yang tidak urgent sama sekali

Dari tweet yang masuk klasifikasi negatif, sebagian besar dapat dikatakan negatif. 10 tweet diatas termasuk dalam kategori negatif dan memang terdengar seperti narasi negatif.

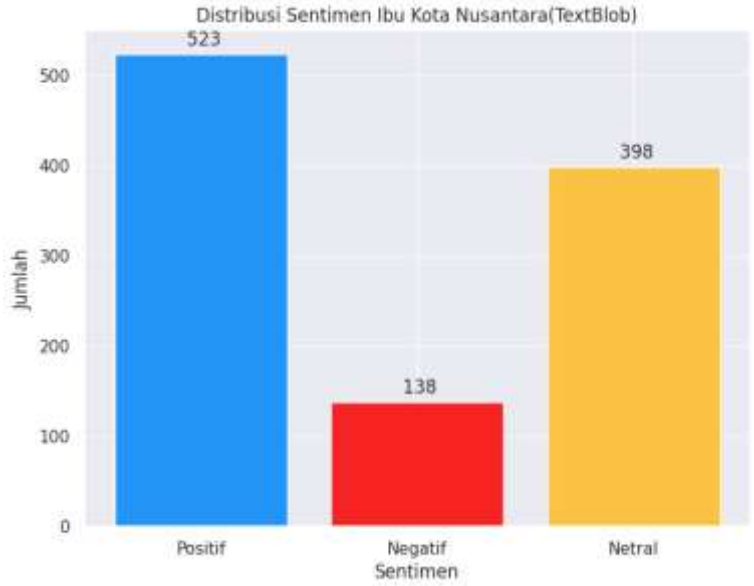
Tabel 3. 3. Tweet netral dengan klasifikasi textblob

10 Tweet dengan klasifikasi netral
airlangga hartarto ungkap kampanye akhir partai golkar bakal gelar ikn
otorita ikn revitalisasi 5 sekolah ikn nusantara

sama2 satu tuju cuan ikn
ikn nusantara jadi tuju investasi dunia
hidmat tuhan ikn
pak bas engga mau bawa aku ikn ya
ohh investor ikn ya
bangun ikn prabowo klaim dukung penuh suku dayak
pak ganjar komitmen ikn
fbr dukung lanjut ikn lewat ganjar-mahfud md

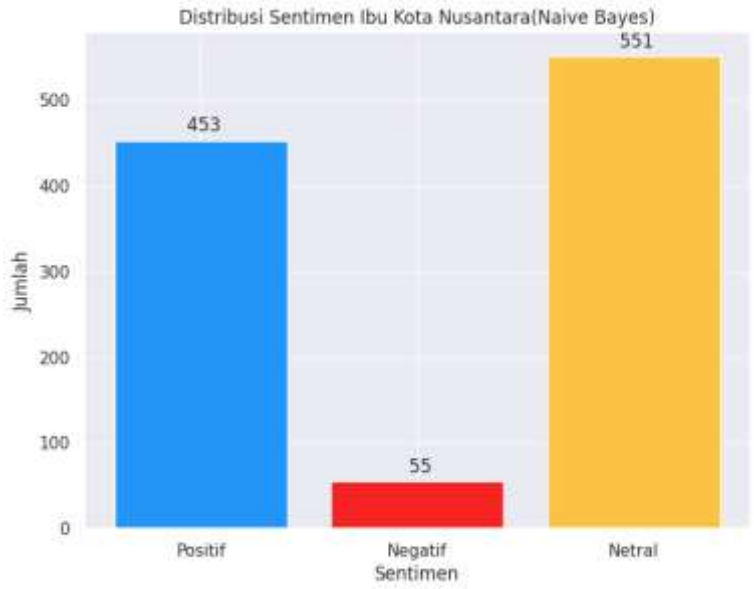
Dari tweet yang masuk klasifikasi netral, sebagian besar dapat dikatakan netral. 10 tweet diatas termasuk dalam kategori netral dan memang terdengar seperti narasi netral.

Setelah kami membuat wordcloud, kami membagi menjadi tiga kelompok guna mengetahui kelompok sentimen dalam proyek IKN. Adapun pembagian sentimen yang kami lakukan yaitu positif, negatif, dan netral. Dilanjutkan dengan membuat *plot* dengan textblob yang bertujuan untuk melihat jumlah masyarakat yang terklasifikasi memiliki sentimen positif, negatif, atau netral terhadap proyek IKN. Hasil yang kami dapatkan setelah melakukan ploting dapat dilihat pada Gambar 3.5. Terlihat bahwa tingkat sentimen masyarakat terhadap proyek IKN lebih condong ke arah positif dengan masa sebanyak 523 orang, sedangkan peringkat kedua diduduki oleh masyarakat yang memiliki sentimen netral dengan masa sebanyak 398 orang, dan terakhir terdapat 138 orang masyarakat yang memiliki sentimen negatif terhadap proyek IKN.



Gambar 3. 5. Distribusi sentimen IKN (TextBlob)

Setelah melakukan pengolahan data menggunakan algoritma Naive Bayes didapati hasil yang berbeda dari sebelumnya. Terlihat pada Gambar 3.6, tingkat sentimen masyarakat terhadap proyek IKN sekarang lebih condong ke arah netral dengan masa sebanyak 551 orang yang sebelumnya 398 orang atau mengalami peningkatan sekitar 40% dan kelompok masyarakat dengan sentimen netral mendominasi dari yang sebelumnya didominasi oleh sentimen positif. Peringkat kedua diduduki oleh masyarakat yang memiliki sentimen positif terhadap proyek IKN dengan jumlah 453 orang yang sebelumnya 523 atau mengalami penurunan. Peringkat terakhir yaitu masyarakat dengan sentimen negatif dengan jumlah 55 orang yang sebelumnya 138 orang.



Gambar 3. 6. Distribusi sentimen IKN (Naive Bayes)

Kami melakukan plotting cloudword untuk setiap klasifikasi sentimen masyarakat terhadap proyek IKN. Pada Gambar 3.7 merupakan hasil cloudword berdasarkan algoritma Naive Bayes terhadap masyarakat yang termasuk ke dalam kelompok yang memiliki sentimen positif terhadap proyek IKN.

Gambar 3. 7. Cloudword klasifikasi bayes positif

Selanjutnya kami membuat wordcloud guna mengetahui kata yang sering digunakan dan masuk ke dalam kelompok yang memiliki sentimen negatif yang dapat dilihat pada Gambar 3.8.

Gambar 3. 8. Cloudword klasifikasi bayes negatif

Distribusi sentimen IKN (Naive Bayes)Terakhir, kami membuat wordcloud untuk mengetahui kata yang sering digunakan oleh masyarakat dan masuk ke dalam klasifikasi sentimen netral terhadap proyek IKN yang dapat dilihat pada Gambar 3.9.

Gambar 3. 9. Cloudword klasifikasi bayes netral

Pada Tabel 3.4 berisikan tweet dari masyarakat yang tergolong ke dalam klasifikasi sentimen positif berdasarkan algoritma Naive Bayes. Terdapat 5 tweet yang penulis pilih dari klasifikasi sentimen positif. Terlihat beberapa kata yang menandakan kalimat tersebut diklasifikasikan menjadi positif, yaitu kata "bangun". Pada data 5 tweet yang dipilih juga tidak sepenuhnya positif sekalipun berdasarkan algoritma Naive Bayes tergolong positif, seharusnya tweet tersebut tidak masuk ke dalam sentimen positif dan tergolong sentimen negatif, tetapi karena machine learning yang digunakan belum akurat dan sangat sulit untuk menjadi akurat, sehingga pasti terjadi kesalahan.

Tabel 3. 4. Tweet possitig dengan klasifikasi naive bayes

5 Tweet dengan klasifikasi positif berdasarkan algoritma Naive Bayes
bismillah aww yuk investor partisipasi doa usaha dukung kembang unggul kalimantan timur jadi ikn segala baik investor milik tuhan yang mahaesa segingga wujud yang amanat dahulu aamiin aww
smg ikn lanjut pak ganjar 2 kali dukung bapak kali harus dukung ganjar akhir jabat bapak nampak jalan spt dulu sangat sayang
sempat plin plan terus balik dukung ikn kek warga takut plin plan
segala dampak positif hasil bangun ikn nusantara harap seluruh elemen masyarakat turut dukung langsung bangun maju indonesia
rakyat pulau kalimantan dukung ikn arti rakyat nya terasut kelompok elit elit jahil ada pusat pulau jawa bangga 78 thn indonesia merdeka jadi pulau kalimantan jadi ikn indonesia timur thn 2024 hutri 79 upacara ikn

Kami memilih 5 tweet dengan klasifikasi negatif berdasarkan algoritma Naive Bayes yang tertuang dalam Tabel 3.5. Adapun kata yang terindikasi menjadi indikator pengelompokan sentimen

negatif yaitu pada kata “proyek”, karena berdasarkan 5 contoh yang diambil, semua terdengar negatif dan di dalamnya terkandung kata “proyek”.

Tabel 3. 5. Tweet negatif dengan klasifikasi naive bayes

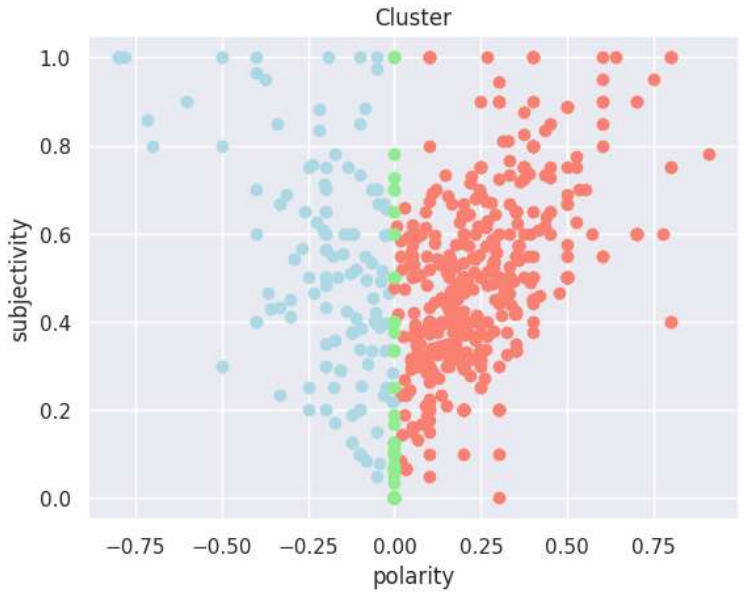
5 Tweet dengan klasifikasi negatif berdasarkan algoritma Naive Bayes
ikn unfaeda dapat faedah prabowo sama hasyim punya proyek besar ikn
dapet proyek ikn bilang hikmat tuhan perkara duniawi tuhan slalu bawa
proyek bangun ikn nusantara serap tenaga kerja
dapet proyek ikn bilang hikmat tuhan perkara duniawi tuhan slalu baw
proyek bangun ikn nusantara serap 9 976 tenaga kerja 30 antara kerja lokal

Kemudian kami memilih juga 5 tweet dengan klasifikasi netral berdasarkan algoritma Naive Bayes. Pada Tabel 3.6 terlihat penggunaan kata “nusantara” terindikasi menjadi indikator pengelompokan sentimen netral.

Tabel 3. 6. Tweet netral dengan klasifikasi naive bayes

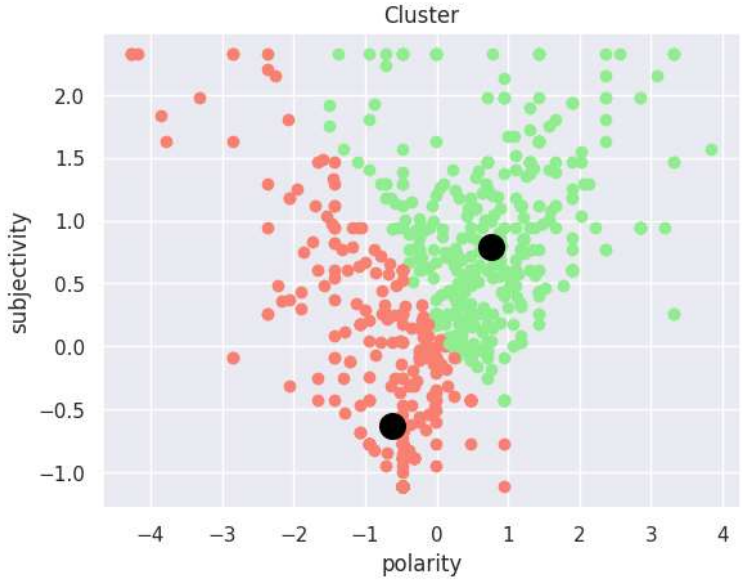
5 Tweet dengan klasifikasi netral berdasarkan algoritma Naive Bayes
ikn nusantara tingkat ekonomi wilayah
meumum rumah rakyat pupr bangun memorial park ibu kota nusantara ikn kalimantan timur bangun memorial park guna hormat jasa pahlawan bangsa nana mirdad bahrain bal bank mandiri yessica tamara
proyek bangun ikn nusantara serap 9 976 tenaga kerja 30 antara kerja lokal
ikn nusantara jadi tuju investasi dunia
bangun ikn nusantara melindugi lestari hutan

Distribusi persebaran data sentimen ketika divisualisasikan dengan scatter terlihat pada gambar 3.10. Warna biru menandakan sentimen negatif, warna hijau untuk sentimen netral dan warna merah untuk sentimen positif.



Gambar 3. 10. Visualisasi persebaran data positif, negatif, dan netral

Distribusi persebaran data K-Means ketika divisualisasikan dengan scatter terlihat pada Gambar 3.11. Data diklasifikasikan kedalam 2 centroid.



Gambar 3. 11. Visualisasi K-means dengan 2 centroid

Untuk memudahkan pembacaan dari visualisasi K-Means dengan 2 centroid, kami melakukan visualisasi untuk cloudword pada setiap centroid.

Gambar 3. 12. Cloudword k-means 2 centroid dengan data berwarna merah

Gambar 3. 13. Cloudword k-means 2 centroid dengan data berwarna hijau

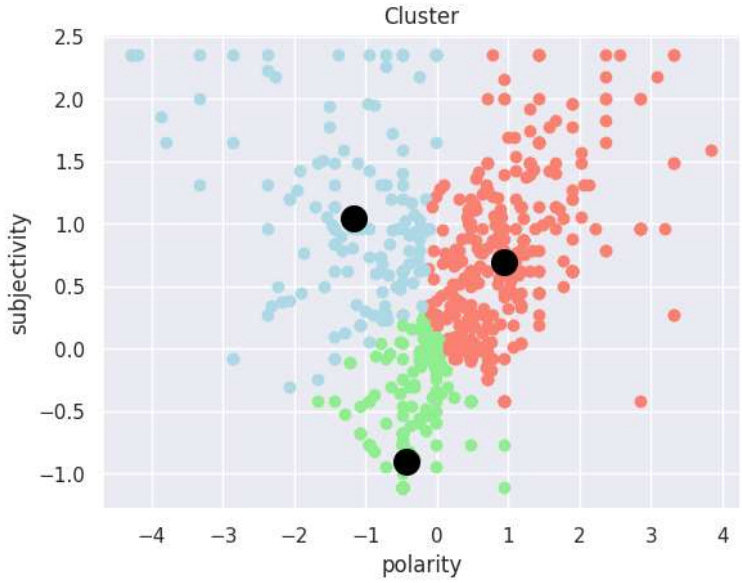
Distribusi persebaran data K-Means ketika divisualisasikan dengan scatter terlihat pada gambar 3.14. Data diklasifikasikan kedalam 3 centroid.

Berdasarkan lokasi centroid dan posisi warna dalam gambar 3.14, dapat dilihat bahwa centoid dengan data berwarna hijau sebagian besar berada dititik polarity nol, yang dapat diartikan bahwa data tweet yang ada terklarifikasi netral. Sebagian besar data dengan warna hijau juga berada dibawah subjectifity 0 yang dapat diartikan sebagai data tweet yang ada di centroid tersebut lebih mendekati opini.

Berdasarkan lokasi centroid dan posisi warna dalam gambar 3.14, dapat dilihat bahwa centoid dengan data berwarna merah sebagian besar berada dititik polarity lebih besar daripada nol, yang dapat diartikan bahwa data tweet yang ada terklarifikasi positif. Sebagian besar data dengan warna hijau juga berada diatas subjectifity 0 yang dapat diartikan sebagai data tweet yang ada di centroid tersebut lebih mendekati faktual.

Berdasarkan lokasi centroid dan posisi warna dalam gambar 3.14, dapat dilihat bahwa centoid dengan data berwarna biru sebagian besar berada dititik polarity lebih kecil daripada nol, yang dapat diartikan bahwa data tweet yang ada terklarifikasi negatif. Sebagian besar data dengan

warna hijau juga berada diatas subjectifity 0 yang dapat diartikan sebagai data tweet yang ada di centroid tersebut lebih mendekati faktual.



Gambar 3. 14. Visualisasi K-means dengan 3 centroid

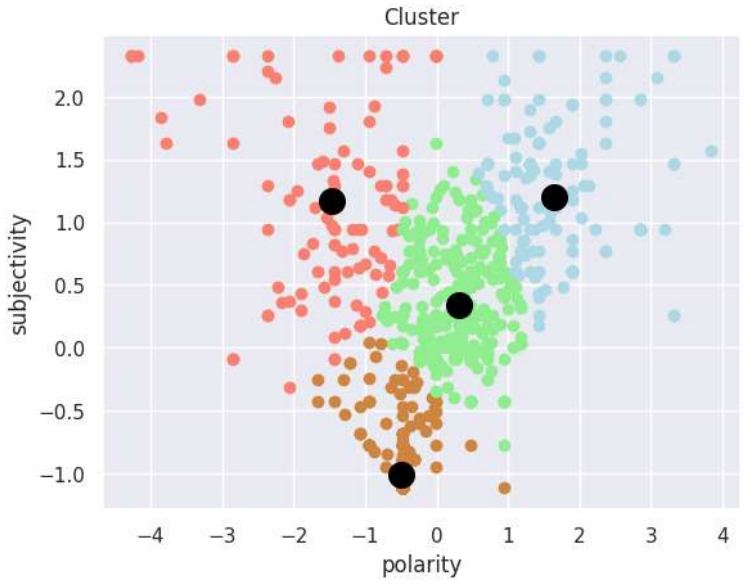
Untuk memudahkan pembacaan dari visualisasi K-Means dengan 3 centroid, kami melakukan visualisasi untuk cloudword pada setiap centroid.

Gambar 3. 15. Cloudword k-means 3 centroid dengan data berwarna merah

Gambar 3. 16. Cloudword k-means 3 centroid dengan data berwarna hijau

Gambar 3. 17.. Cloudword k-means 3 centroid dengan data berwarna biru

Distribusi persebaran data K-Means ketika divisualisasikan dengan scatter terlihat pada Gambar 3.11. Data diklasifikasikan kedalam 4 centroid.



Gambar 3. 18. . Visualisasi K-means dengan 4 centroid

Untuk memudahkan pembacaan dari visualisasi K-Means dengan 3 centroid, kami melakukan visualisasi untuk cloudword pada setiap centroid.

Gambar 3. 19. Cloudword k-means 4 centroid dengan data berwarna merah

Gambar 3. 20. Cloudword k-means 4 centroid dengan data berwarna hijau

Gambar 3. 21. Cloudword k-means 4 centroid dengan data berwarna biru

Gambar 3. 22. . Cloudword k-means 4 centroid dengan data berwarna coklat

BAB IV. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Melalui hasil pengolahan data yang kami lakukan, dapat ditarik kesimpulan bahwa, pandangan masyarakat serta opini masyarakat dan sentimen masyarakat terhadap proyek IKN cenderung ke arah netral. Hal ini mungkin disebabkan oleh opini tokoh masyarakat, pemerintah, maupun informasi yang didapatkan oleh masyarakat

5.2. Saran

Apabila terdapat kekurangan dalam laporan yang dibuat mohon untuk diinformasikan kepada penulis agar bisa dijadikan bahan perbaikan untuk kedepannya.

DAFTAR PUSTAKA

Affandi, Y., & Sugiharti, E. (2023). Sentiment Analysis of Student on Online Lectured During Covid-19 Pandemic Using K-Means and Naïve Bayes Classifier ARTICLE INFO ABSTRACT. Journal of Advances in Information Systems and Technology, 5(1).

Majid, A., Nugraha, D., & Dharma Adhinata, F. (2023). Sentiment Analysis on Tiktok Application Reviews Using Natural Language Processing Approach. https://doi.org/10.26858/jessi.v4i1.41897

Nainggolan, R., Adline, F., Tobing, T., & Harianja, E. J. G. (2022). Analysis Sentiment in Bukalapak Comments with K-Means Clustering Method. International Journal of New Media Technology), 9(2), 87.

Sinaga, R. B., Al Fajri, H. R., Mubarok, H., Pangestu, A. D., & Prasvita, D. S. (2021). Analisis Sentimen Pengguna Twitter terhadap Konflik antara Palestina dan Israel Menggunakan Metode Naïve Bayesian Classification dan Support Vector Machine. In Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya (Vol. 2, No. 2, pp. 166-175).

LAMPIRAN

1. Code

```
# -*- coding: utf-8 -*-
"""Kode_Tubes_PSD

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1gjxdfk5TDwe6sstTknANbYCN0qOiVxJa

# ***CRAWLING DATA X KEYWORD : IKN***
"""

!pip install pandas
!curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -
!sudo apt-get install -y nodejs

import pandas as pd

data = 'data_ikn.csv'
search_keyword = 'ikn until:2024-01-21 since:2023-12-22 lang:id'
limit = 2000

!npx --yes tweet-harvest@2.2.8 -o "{data}" -s "{search_keyword}" -l {limit} --token ""

data = pd.read_csv("/content/tweets-data/data_ikn.csv", sep=";")

data.info()

data.head()

"""# ***CRAWLING DATA X KEYWORD : #IKN***"""

data = 'data_ikn_hastag.csv'
search_keyword = '(#ikn) until:2024-01-21 since:2023-12-22 lang:id'
limit = 2000

!npx --yes tweet-harvest@2.2.8 -o "{data}" -s "{search_keyword}" -l {limit} --token ""

data = pd.read_csv("/content/tweets-data/data_ikn_hastag.csv", sep=";")

data.info()

data.head()

"""#penggabungan data"""

# Baca dua file CSV
file1 = pd.read_csv('/content/tweets-data/data_ikn.csv', sep=';')
file2 = pd.read_csv('/content/tweets-data/data_ikn_hastag.csv', sep=';')

# Gabungkan dua file
merged_data = pd.concat([file1, file2], ignore_index=True)
```

```
# Simpan hasil gabungan ke file CSV baru
merged_data.to_csv('/content/tweets-data/data_ikn_scraping.csv', sep=';', index=False)

"""# Nampilin data"""

import pandas as pd
import re
import seaborn as sns
import matplotlib.pyplot as plt

data = pd.read_csv("/content/tweets-data/data_ikn_scraping.csv", sep=";")

data = data[['full_text', 'username', 'created_at']]
data

"""# Cleansing data"""

data.shape

data = data.drop_duplicates(subset=['full_text'])

data.duplicated().sum()

data = data.dropna()

data.isnull().sum()

data.shape

def clean_twitter_text(text):
  text = re.sub(r'@[A-Za-z0-9_]+', ' ', text)
  text = re.sub(r'#\w+',' ', text)
  text = re.sub(r'RT[\s]+', ' ', text)
  text = re.sub(r'https?://\S+', ' ', text)
  text = re.sub(r' [^A-Za-z0-9()]', ' ', text)
  text = re.sub(r'\s+', ' ', text).strip()
  text = re.sub(r'\(|\)', ' ', text)
  text = re.sub(r"[.,]", ' ', text)

  return text

data['full_text'] = data['full_text'].apply(clean_twitter_text)

def clean_twitter_space(text):
  text = re.sub(r'\s+', ' ', text).strip()

  return text

data['full_text'] = data['full_text'].apply(clean_twitter_space)

data['full_text'] = data['full_text'].str.lower()

data
```

```
"""# Preprocessing"""

#normalisasi
norm = {" yg ": " yang ", " tdk ": " tidak ", " utk ": " untuk ", " ni ": " ini ", " aja ": " saja ",
    " klo ": " kalau ", " krn ": " karena ", " ga ": " tidak ", " dgn ": " dengan ", " jd ": " jadi ",
    " bgt ": " banget ", " gpp ": " tidak apa-apa ", " dr ": " dari ", " lg ": " lagi ", " yaa ": " ya ",
    " gt ": " begitu ", " sllu ": " selalu ", " sgt ": " sangat ", " sm ": " sama ", " kok ": " kenapa ",
    " bgd ": " banget ", " gw ": " saya ", " lo ": " kamu ", " gak ": " tidak ", " krja ": " kerja ",
    " slmt ": " selamat ", " bsk ": " besok ", " km ": " kamu ", " mw ": " mau ", " dpt ": " dapat ",
    " org ": " orang ", " krg ": " kurang ", " ama ": " sama ", " tu ": " itu ", " laen ": " lain ",
    " ntar ": " sebentar ", " sempet ": " sempat ", " emg ": " memang ", " tp ": " tapi ", " lu ": " kamu ",
    " smpe ": " sampai ", " yha ": " ya ", " bener ": " benar ", " brp ": " berapa ", " byk ": " banyak ",
    " gmn ": " bagaimana ", " bs ": " bisa ", " dy ": " dia ", " proy3k ": " proyek ", " odong2an ": " odong-odongan "}

def normalisasi(str_text):
  for i in norm:
    str_text = str_text.replace(i, norm[i])
  return str_text

data['full_text'] = data['full_text'].apply(lambda x: normalisasi(x))
data

!pip install sastrawi

## Stopwords
import Sastrawi

from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory, StopWordRemover, ArrayDictionary
more_stop_words = ["yang", "bahwa", "itu", "ini", "dengan", "untuk", "pada", "ke", "oleh", "dari", "sebagai", "atau", "kepada", "namun", "demikian", "sebagaimana", "meskipun", "jika", "karena", "sejak", "dan", "tetapi", "melainkan", "sementara", "ialah", "adalah", "artinya", "yaitu", "misalnya", "contohnya", "pada", "saat", "sekarang", "akan", "sudah", "belum", "lagi", "sudah", "pernah", "sering", "setiap", "semua", "beberapa", "banyak", "sedikit", "hampir", "hingga", "sampai", "di", "dalam", "luar", "dari", "ke", "oleh", "pada", "dengan", "tanpa", "untuk", "seperti", "bagaimana", "mengapa", "kapan", "apa", "siapa", "mana", "dari", "ke", "oleh", "pada", "dengan", "yang", "ini", "itu", "adalah", "agar", "supaya", "sehingga", "untuk", "dari", "ke", "oleh", "pada", "dengan", "karena", "supaya", "bahwa", "agar", "biar", "supaya"]

stop_words = StopWordRemoverFactory().get_stop_words()
stop_words.extend(more_stop_words)

new_array = ArrayDictionary(stop_words)
```

```
stop_words_remover_new = StopWordRemover(new_array)

def stopword(str_text):
    str_text = stop_words_remover_new.remove(str_text)
    return str_text

data['full_text'] = data['full_text'].apply(lambda x: stopword(x))

data

#tokenize
tokenized = data['full_text'].apply(lambda x:x.split())
tokenized

from google.colab import drive
drive.mount('/content/drive')

# Tokenization and Stemming
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from Sastrawi.StopWordRemover.StopWordRemoverFactory import
StopWordRemoverFactory, StopWordRemover, ArrayDictionary

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

def tokenize_and_stem(text):
    tokens = text.split()
    stemmed_tokens = [stemmer.stem(token) for token in tokens]
    return ' '.join(stemmed_tokens)

data['tokenized_and_stemmed'] = data['full_text'].apply(tokenize_and_stem)

# Save the preprocessed data to a CSV file
data.to_csv("/content/drive/My Drive/UAS PSD/data_ikn_preprocessing.csv",
index=False)

if 'created_at' in data.columns:
    data = data.drop(columns=['username', 'created_at', 'full_text'])

# Save the preprocessed data to a CSV file
data.to_csv("/content/drive/My Drive/UAS PSD/data_ikn_preprocessing.csv",
index=False)

data

!pip install mtranslate==1.8

import pandas as pd
from mtranslate import translate

data = pd.read_csv("/content/drive/My Drive/UAS PSD/data_ikn_preprocessing.csv")

def convert_eng(tweet):
    translation = translate(tweet, "en", "id")
    return translation
```

```
data.info()

data['tweet_english'] = data['tokenized_and_stemmed'].apply(convert_eng)

data.to_csv("/content/drive/My Drive/UAS PSD/data_ikn_translate.csv", index=False)

data

"""#Labeling"""

!pip install tweet-preprocessor
!pip install textblob
!pip install wordcloud
!pip install nltk

import preprocessor as p
from textblob import TextBlob
import nltk
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize

nltk.download('punkt')

data = pd.read_csv("/content/drive/My Drive/UAS PSD/data_ikn_translate.csv")

data['tweet_english'] = data['tweet_english'].str.lower()
data

data_tweet = list(data['tweet_english'])
polaritas = 0

status = []
total_positif = total_negatif = total_netral = total = 0

# Iterasi setiap tweet
for i, tweet in enumerate(data_tweet):
  # Lakukan sentiment analysis
  analysis = TextBlob(tweet)
  polaritas += analysis.polarity
  # Hitung jumlah positif, negatif, netral, dan total tweet
  if analysis.sentiment.polarity > 0.0:
    total_positif += 1
    status.append('Positif')
  elif analysis.sentiment.polarity == 0.0:
    total_netral += 1
    status.append('Netral')
  else:
    total_negatif += 1
    status.append('Negatif')

  total += 1

# Cetak hasil
print('Jumlah tweet positif:', total_positif)
```

```
print('Jumlah tweet negatif:', total_negatif)
print('Jumlah tweet netral:', total_netral)
print('Jumlah total tweet:', total)

data['klasifikasi'] = status

"""#Visualisasi"""

import matplotlib.pyplot as plt
import numpy as np

from wordcloud import WordCloud, STOPWORDS

def plot_cloud(wordcloud):
    plt.figure(figsize=(10, 8))
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis('off')
    plt.show()

all_words = ' '.join([tweets for tweets in data['tokenized_and_stemmed']])

wordcloud = WordCloud(
    width=3000,
    height=2000,
    random_state=3,
    background_color='black',
    colormap='Blues_r',
    collocations=False,
    stopwords=STOPWORDS
).generate(all_words)

plot_cloud(wordcloud)

positif_words = ' '.join([tweets for tweets in data[data['klasifikasi'] ==
'Positif']['tokenized_and_stemmed']])

wordcloud_positif = WordCloud(
    width=3000,
    height=2000,
    random_state=3,
    background_color='black',
    colormap='Blues_r',
    collocations=False,
    stopwords=STOPWORDS
).generate(positif_words)

plot_cloud(wordcloud_positif)

negatif_words = ' '.join([tweets for tweets in data[data['klasifikasi'] ==
'Negatif']['tokenized_and_stemmed']])

wordcloud_negatif = WordCloud(
    width=3000,
    height=2000,
    random_state=3,
```

```
    background_color='black',
    colormap='Blues_r',
    collocations=False,
    stopwords=STOPWORDS
).generate(negatif_words)

plot_cloud(wordcloud_negatif)

netral_words = ' '.join([tweets for tweets in data[data['klasifikasi'] ==
'Netral']['tokenized_and_stemmed']])

wordcloud_netral = WordCloud(
    width=3000,
    height=2000,
    random_state=3,
    background_color='black',
    colormap='Blues_r',
    collocations=False,
    stopwords=STOPWORDS
).generate(netral_words)

plot_cloud(wordcloud_netral)

data_positif = data[data['klasifikasi'] == 'Positif']

# Menampilkan 10 data secara lengkap
print(data_positif['tokenized_and_stemmed'].head(10))
data_positif.to_csv('hasil_filter_positif.csv', index=False)

data_negatif = data[data['klasifikasi'] == 'Negatif']

# Menampilkan 10 data secara lengkap
print(data_negatif['tokenized_and_stemmed'].head(10))
data_negatif.to_csv('hasil_filter_negatif.csv', index=False)

data_netral = data[data['klasifikasi'] == 'Netral']

# Menampilkan 10 data secara lengkap
print(data_netral['tokenized_and_stemmed'].head(10))
data_netral.to_csv('hasil_filter_netral.csv', index=False)

import seaborn as sns

sns.set_theme()

labels = ['Positif', 'Negatif', 'Netral']
counts = [total_positif, total_negatif, total_netral]

def show_bar_chart(labels, counts, title):
    fig, ax = plt.subplots(figsize=(8, 6))
    bars = ax.bar(labels, counts, color=['#2394f7', '#f72323', '#fac343'])

    for bar, count in zip(bars, counts):
        height = bar.get_height()
        ax.annotate(f' {count}', xy=(bar.get_x() + bar.get_width() / 2, height),
```

```
                xytext=(0, 3),
                textcoords="offset points",
                ha='center', va='bottom')

    ax.grid(axis='y', linestyle='-', alpha=0.7)

    ax.set_xlabel('Sentimen')
    ax.set_ylabel('Jumlah')

    ax.set_title(title)

    plt.show()

show_bar_chart(labels, counts, "Distribusi Sentimen Ibu Kota Nusantara(TextBlob)")

"""#klasifikasi Sentimen"""

data

selected_columns = ['tweet_english', 'klasifikasi']
selected_data = data[selected_columns]

dataset = [tuple(x) for x in selected_data.to_records(index=False)]

dataset

import random

set_positif = []
set_negatif = []
set_netral = []

for n in dataset:
    if n[1] == 'Positif':
        set_positif.append(n)
    elif n[1] == 'Negatif':
        set_negatif.append(n)
    else:
        set_netral.append(n)

set_positif = random.sample(set_positif, k=int(len(set_positif)/2))
set_negatif = random.sample(set_negatif, k=int(len(set_negatif)/2))
set_netral = random.sample(set_netral, k=int(len(set_netral)/2))

train = set_positif + set_negatif + set_netral

train_set = []

for n in train:
    train_set.append(n)

from textblob.classifiers import NaiveBayesClassifier
cl = NaiveBayesClassifier(train_set)
print("Akurasi Test: ", cl.accuracy(dataset))
```

```
# Labeling
data_tweet = list(data['tweet_english'])
polaritas = 0

status = []
total_positif = total_negatif = total_netral = total = 0

for i, tweet in enumerate(data_tweet):
    analysis = TextBlob(tweet, classifier=cl)

    if analysis.classify() == 'Positif':
        total_positif += 1
    elif analysis.classify() == 'Netral':
        total_netral += 1
    else:
        total_negatif += 1

    status.append(analysis.classify())
    total += 1

print('\nHasil Analisis Data:\nPositif = {}\nNetral = {}\nNegatif =
{}'.format(total_positif, total_netral, total_negatif))
print("\nTotal Data: {}".format(total))

status = pd.DataFrame({"Klasifikasi Bayes": status})
data['klasifikasi bayes'] = status

sns.set_theme()

labels = ['Positif', 'Negatif', 'Netral']
counts = [total_positif, total_negatif, total_netral]

def show_bar_chart(labels, counts, title):
    fig, ax = plt.subplots(figsize=(8, 6))
    bars = ax.bar(labels, counts, color=['#2394f7', '#f72323', '#fac343'])

    for bar, count in zip(bars, counts):
        height = bar.get_height()
        ax.annotate(f' {count}', xy=(bar.get_x() + bar.get_width() / 2, height),
                    xytext=(0, 3),
                    textcoords="offset points",
                    ha='center', va='bottom')

    ax.grid(axis='y', linestyle='-', alpha=0.7)

    ax.set_xlabel('Sentimen')
    ax.set_ylabel('Jumlah')

    ax.set_title(title)

    plt.show()

show_bar_chart(labels, counts, "Distribusi Sentimen Ibu Kota Nusantara(Naive
Bayes)")
```

```
data_netral = data[data['klasifikasi bayes'] == 'Netral']

# Menampilkan 10 data secara lengkap
print(data_netral['tokenized_and_stemmed'].head(100))
data_netral.to_csv('hasil_filter_netral.csv', index=False)

data_negatif = data[data['klasifikasi bayes'] == 'Negatif']

# Menampilkan 10 data secara lengkap
print(data_negatif['tokenized_and_stemmed'].head(10))
data_negatif.to_csv('hasil_filter_negatif.csv', index=False)

data_positif = data[data['klasifikasi bayes'] == 'Positif']

# Menampilkan 10 data secara lengkap
print(data_positif['tokenized_and_stemmed'].head(10))
data_positif.to_csv('hasil_filter_positif.csv', index=False)

data

def cetak_data_eval(data):
  data_eval = tuple(x for x in data.to_records(index=False))
  for n in data_eval:
    if len(n) >= 4:
      if n[2] != n[3]:
        print(f'Text: {n[0]}\nClassifier: {n[2]}\nClassifier Bayes: {n[3]} \n')
    else:
      print("Tuple tidak memiliki cukup elemen.")

cetak_data_eval(data)

positif_words = ' '.join([tweets for tweets in data[data['klasifikasi bayes'] ==
'Positif']['tokenized_and_stemmed']])

wordcloud_positif = WordCloud(
    width=3000,
    height=2000,
    random_state=3,
    background_color='black',
    colormap='Blues_r',
    collocations=False,
    stopwords=STOPWORDS
).generate(positif_words)

plot_cloud(wordcloud_positif)

negatif_words = ' '.join([tweets for tweets in data[data['klasifikasi bayes'] ==
'Negatif']['tokenized_and_stemmed']])

wordcloud_negatif = WordCloud(
    width=3000,
    height=2000,
    random_state=3,
    background_color='black',
    colormap='Blues_r',
```

```
    collocations=False,
    stopwords=STOPWORDS
).generate(negatif_words)

plot_cloud(wordcloud_negatif)

netral_words = ' '.join([tweets for tweets in data[data['klasifikasi bayes'] ==
'Netral']['tokenized_and_stemmed']])

wordcloud_netral = WordCloud(
    width=3000,
    height=2000,
    random_state=3,
    background_color='black',
    colormap='Blues_r',
    collocations=False,
    stopwords=STOPWORDS
).generate(netral_words)

plot_cloud(wordcloud_netral)

"""# SUMBER BERBEDA (K-means)"""

def getSubjectivity(text):
    analysis = TextBlob(text)
    return analysis.sentiment.subjectivity

data['subjectivity'] = data['tweet_english'].apply(getSubjectivity)
data

def getPolarity(text):
    analysis = TextBlob(text)
    return analysis.sentiment.polarity

data['polarity'] = data['tweet_english'].apply(getPolarity)
data

positif = data[data["klasifikasi"] == 'Positif']
negatif = data[data["klasifikasi"] == 'Negatif']
netral = data[data["klasifikasi"] == 'Netral']
print("+ " + str((positif.shape[0]/data.shape[0]) * 100 ) + ' %')
print("- " + str((negatif.shape[0]/data.shape[0]) * 100 ) + ' %')
print("~ " + str((netral.shape[0]/data.shape[0]) * 100 ) + ' %')

for index, row in data.iterrows():
    if row['klasifikasi'] == 'Positif':
        plt.scatter(row['polarity'], row['subjectivity'], color="salmon", label='Positif')
    elif row['klasifikasi'] == 'Netral':
        plt.scatter(row['polarity'], row['subjectivity'], color="lightgreen", label='Netral')
    elif row['klasifikasi'] == 'Negatif':
        plt.scatter(row['polarity'], row['subjectivity'], color="lightblue", label='Negatif')

plt.title('Cluster')
plt.xlabel('polarity')
plt.ylabel('subjectivity')
```

```
plt.show()

from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
data[['polarity', 'subjectivity']] = scaler.fit_transform(data[['polarity', 'subjectivity']])

n_clusters = 4

x_array = data[['polarity', 'subjectivity']]

kmeans = KMeans(n_clusters=n_clusters, random_state=42)
data['cluster'] = kmeans.fit_predict(x_array)

data['kluster'] = kmeans.labels_
data

for index, row in data.iterrows():
    if row['kluster'] == 0:
        plt.scatter(row['polarity'], row['subjectivity'], color="salmon")
    elif row['kluster'] == 1:
        plt.scatter(row['polarity'], row['subjectivity'], color="lightgreen")
    elif row['kluster'] == 2:
        plt.scatter(row['polarity'], row['subjectivity'], color="lightblue")
    elif row['kluster'] == 3:
        plt.scatter(row['polarity'], row['subjectivity'], color="peru")

plt.title('Cluster')
plt.xlabel('polarity')
plt.ylabel('subjectivity')
plt.show()

centroids = kmeans.cluster_centers_
print(centroids)

for index, row in data.iterrows():
    if row['kluster'] == 0:
        plt.scatter(row['polarity'], row['subjectivity'], color="salmon")
    elif row['kluster'] == 1:
        plt.scatter(row['polarity'], row['subjectivity'], color="lightgreen")
    elif row['kluster'] == 2:
        plt.scatter(row['polarity'], row['subjectivity'], color="lightblue")
    elif row['kluster'] == 3:
        plt.scatter(row['polarity'], row['subjectivity'], color="peru")

plt.title('Cluster')
plt.xlabel('polarity')
plt.ylabel('subjectivity')
plt.scatter(centroids[:,0],centroids[:,1], color='black', s=200)
plt.show()

data_polarity_0 = data[data['polarity'] == 0]
```

```
text_polarity_0 = ' '.join([tweets for tweets in data['tokenized_and_stemmed']])

wordcloud = WordCloud(
    width=800,
    height=400,
    background_color='black',
    colormap='Blues_r',
).generate(text_polarity_0)

plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('data warna merah')
plt.axis('off')
plt.show()

data_polarity_0 = data[data['polarity'] == 1]

text_polarity_0 = ' '.join([tweets for tweets in data['tokenized_and_stemmed']])

wordcloud = WordCloud(
    width=800,
    height=400,
    background_color='black',
    colormap='Blues_r',
).generate(text_polarity_0)

plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('data warna hijau')
plt.axis('off')
plt.show()

data_polarity_0 = data[data['polarity'] == 2]

text_polarity_0 = ' '.join([tweets for tweets in data['tokenized_and_stemmed']])

wordcloud = WordCloud(
    width=800,
    height=400,
    background_color='black',
    colormap='Blues_r',
).generate(text_polarity_0)

plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('data warna biru')
plt.axis('off')
plt.show()

data_polarity_0 = data[data['polarity'] == 3]

text_polarity_0 = ' '.join([tweets for tweets in data['tokenized_and_stemmed']])

wordcloud = WordCloud(
    width=800,
```

```
    height=400,
    background_color='black',
    colormap='Blues_r',
).generate(text_polarity_0)

plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('data warna coklat')
plt.axis('off')
plt.show()
```

2. Dataset
   Dataset pada tugas besar kali ini: https://drive.google.com/drive/folders/1Wr-pB2TvOXujotnKMgkh0iU8cUHu-D5N

3. Tabel Kontribusi

No	Nama	NIM	Kontibusi
1	Arya Ashari	105221013	• Pengerjaan 90% code • Penyusunan laporan • Membuat ppt
2	David Ephraim	105221002	• Pengerjaan 20% code • Penyusunan laporan • Membuat ppt
3	Christo Zwingly Alexander	105221019	• Penyusunan laporan • Membuat ppt