KLASIFIKASI BOT *TWITTER* DENGAN MENGGUNAKAN *RANDOM FOREST CLASSIFER*

SKRIPSI

Oleh

M. Geofany Hermawan

NIM E41191593

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI JEMBER

2023

KLASIFIKASI BOT *TWITTER* DENGAN MENGGUNAKAN *RANDOM FOREST CLASSIFER*

SKRIPSI

Sebagai salah satu syarat untuk memperoleh gelar Sarjana Sains Terapan Komputer (S. Tr. Kom) di Program Studi Teknik Informatika Jurusan Teknologi Informasi

Oleh

M. Geofany Hermawan

NIM E41191593

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI JEMBER

2023

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI
POLITEKNIK NEGERI JEMBER
JURUSAN TEKNOLOGI INFORMASI

KLASIFIKASI BOT TWITTER DENGAN MENGGUNAKAN RANDOM FOREST CLASSIFIER

M. Geofany Hermawan (E41191593)

Telah Diuji pada Tanggal 26 Mei 2023
Dan dinyatakan Memenuhi Syarat

Ketua Penguji

Prawidya Destarianto, S.kom, M.T
NIP 19801212200501 1 001

Sekretaris Penguji,

Mukhamad Angga Gumilang, S. Pd., M. Eng.
NIP. 19940812 201903 1 013

Anggota Penguji,

Arvita Agus Kurniasari, S.ST.,M.Tr.Kom
NIK. D199308312021032

Dosen Pembimbing

Mukhamad Angga Gumilang, S. Pd., M. Eng.
NIP. 19940812 201903 1 013

Mengesahkan
Ketua Jurusan
Teknologi Informasi

Hendra Yufit Riskiawan, S.Kom, M.Cs
NIP. 19830203 200604 1 003

SURAT PERNYATAAN MAHASISWA

Saya yang bertanda tangan dibawah ini:

Nama : M. Geofany Hermawan

NIM : E41191593

Menyatakan dengan sebenar – benarnya bahwa segala pernyataan yang berada dalam Laporan Akhir atau Skripsi saya yang berjudul ”**KLASIFIKASI BOT TWITTER DENGAN ALGORITMA RANDOM FOREST CLASSIFIER**” merupakan gagasan dan hasil karya saya dengan arahan dari dosen pembimbing, serta belum pernah diajukan dalam bentuk apapun pada perguruan tinggi manapun.

Semua data dan informasi yang digunakan pada laporan ini telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip karya yang telah diterbitkan dari penulisan lain telah disebutkan dalam naskah dan dicantumkan dalam Daftar Pustaka di bagian akhir Laporan Akhir atau Skripsi ini.

Jember, 26 Mei 2023

M. Geofany Hermawan

NIM. E41191593

PERNYATAAN
PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS

Yang bertanda tangan dibawah ini, saya :

Nama	: M. Geofany Hermawan
NIM	: E4119593
Program Studi	: Teknik Informatika
Jurusan	: Teknologi Informasi

Demi pengembangan ilmu pengetahuan, saya menyetujui untuk memberikan kepada UPT Perpustakaan Politeknik Negeri Jember, Hak Bebas Royali Non-Eksklusif (Non -Exclusive Royalty Free Right) atas Karya Ilmiah berupa **Laporan Skripsi** saya yang berjudul :

KLASIFIKASI BOT TWITTER MENGGUNAKAN RANDOM FOREST CLASSIFIER

Dengan hak bebas royalty Non-Eksklusif ini UPT. Perpustakaan Politeknik Negeri Jember berhak menyimpan, mengalih media atau format, mengelola dalam bentuk Pangkalan Data (Database), mendistribusikan karya dan menampilkan atau mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta.

Saya bersedia untuk menanggung secara pribadi tanpa melibatkan pihak Politeknik Negeri Jember. Segala bentuk tuntutan hukum yang timbul atas Pelanggaran Hak Cipta dalam Karya Ilmiah ini

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat Di : Jember
Pada Tanggal : 16 Juni 2023
Yang Menyatakan,

(M. Hermawan)
NIM. E4119593

MOTTO

"Kekuatan terbesar adalah mampu mengendalikan pikiran dan meraih kesadaran diri yang mendalam"

(Henry Maanampiring)

"Kendalikanlah apa yang dapat kamu kendalikan, terimalah dengan bijaksana apa yang tidak dapat kamu kendalikan."

(Filsuf Stoic)

HALAMAN PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Puji syukur kepada Allah SWT yang telah melimpahkan rahmat, karunia serta hidayahNya sehingga penulis dapat menyelesaikan tugas akhir ini dengan tepat waktu dan sesuai dengan yang diharapkan.
2. Kedua orang tua tercinta yang selalu memberi dukungan dan kasih sayang mereka sehingga dapat memberikan sebuah dorongan dan semangat serta keyakinan kepada penulis sehingga bias menyelesaikan tugas akhir ini.
3. Dosen Pembimbing pak Mukhamad Angga Gumilang, S. Pd., M. Eng yang telah memberikan semangat dan bimbingan kepada penulis dalam menyelesaikan skripsi saya.
4. Terima kasih untuk semua dosen Dosen D-IV Teknik Informatik beserta staf karyawan Politeknik Negeri Jember, terima kasih atas semua bimbingan, ilmu dan bantuannya sehingga penulis dapat menyelesaikan skripsi ini dengan maksimal.
5. Terima kasih untuk teman-teman Teknik Informatika Angkatan 2019 atas dukungan dan semangat selama ini.
6. Terima kasih untuk Jurusan Teknologi Informasi dan almamater saya Politeknik Negeri Jember.
7. Semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu penulis dalam menyelesaikan skripsi ini.

Klasifikasi BOT Twitter Dengan Menggunakan *Random Forest Classifier*

Dibimbing oleh Mukhamad Angga Gumilang, S. Pd., M. Eng

M. Geofany Hermawan

Program Studi Teknik Informatika

Jurusan Teknologi Informasi

ABSTRAK

Twitter menjadi media sosial yang populer dan banyak di pakai oleh kalangan milenial ataupun dewasa. Persentase pengguna *Twitter* di Indonesia dari tahun ke tahun juga terus mengalami peningkatan yang pesat. Pada tahun 2019, pengguna *Twitter* di Indonesia bahkan mencapai 6,43 juta pengguna atau sekitar 52% dari total pengguna media sosial. Namun, *Twitter* memiliki tingkat kejahatan tinggi karena banyak pihak-pihak yang tidak bertanggung jawab dengan membuat pengguna akun yang palsu untuk memberikan informasi yang akan membahayakan pengguna lainnya. Oleh karena itu, penulis membuat penelitian dengan menggunakan *Random Forest Classifier* untuk mengetahui sebuah akun pada *Twitter* dinyatakan bot atau *human*. Pada penelitian ini peneliti menggunakan 1200 data sebagai data latih dan 10 data uji. Berdasarkan data latih yang di tersebut dilakukan klasifikasi menggunakan metode *Random Forest Classifier* dan dilakukan pengujian terhadap data testing. Berdasarkan hasil yang dilakukan akurasi tertinggi di dapatkan pada perbandingan data 8:2 dengan akurasi 85%, sedangkan untuk perbandingan data 9:1 di dapat dengan akurasi 81%, data dengan perbandingan 7:3 di dapat dengan akurasi 83%, data dengan perbandingan 6:4 di dapat dengan akurasi 82%, dan data dengan perbandingan 5:5 di dapat dengan akurasi 81%. Dapat di simpulkan bahwa pada penelitian ini data yang digunakan cukup penting hingga dapat mempengaruhi hasil akurasi pada algoritma yang digunakan. Untuk penelitian selanjutnya dibutuhkan data yang lebih banyak untuk menambah tingkat akurasi.

Kata Kunci : *Classification, Data Mining, Twitter, Bot, Random Forest Classifier.*

Twitter Bot Classification Using the Random Forest Classifier Algorithm

M. Geofany Hermawan
Study Program of Informatics Engineering
Majoring in Information Technology

ABSTRACT

Twitter has become a popular social media platform widely used by millennials and adults. The percentage of Twitter users in Indonesia has also been rapidly increasing over the years. In 2019, the number of Twitter users in Indonesia reached 6.43 million or approximately 52% of the total social media users. However, Twitter has a high crime rate due to the presence of irresponsible individuals creating fake accounts to disseminate harmful information to other users. Therefore, the author conducted a research using Random Forest Classifier to determine whether a Twitter account is a bot or a human. In this study, the author used 1200 data as training and 10 testing data. Based on the labeled training data, classification was performed using the Random Forest Classifier method, and testing was conducted using the testing data. The results showed that the highest accuracy was achieved with an 8:2 data ratio, with an accuracy of 85%. The accuracy for the 9:1 data ratio was 81%, while for the 7:3 data ratio, it was 83%. The accuracy for the 6:4 data ratio was 82%, and for the 5:5 data ratio, it was 81%. In conclusion, the data used in this study was found to be significant as it influenced the accuracy results of the algorithm used. For future research, a larger dataset is needed to further improve the accuracy level.

Keywords : *Classification, Data Mining, Twitter, Bot, Random Forest Classifier.*

RINGKASAN

Twitter menjadi media sosial yang populer dan banyak di pakai oleh kalangan milenial ataupun dewasa. Persentase pengguna *Twitter* di Indonesia dari tahun ke tahun juga terus mengalami peningkatan yang pesat. Pada tahun 2019, pengguna *Twitter* di Indonesia bahkan mencapai 6,43 juta pengguna atau sekitar 52% dari total pengguna media sosial. Namun, *Twitter* memiliki tingkat kejahatan tinggi karena banyak pihak-pihak yang tidak bertanggung jawab dengan membuat pengguna akun yang palsu untuk memberikan informasi yang akan membahayakan pengguna lainnya. Oleh karena itu, penulis membuat penelitian dengan menggunakan *Random Forest Classifier* untuk mengetahui sebuah akun pada *Twitter* dinyatakan bot atau *human*. Pada penelitian ini peneliti menggunakan 1200 data sebagai data latih dan 10 data uji. Berdasarkan data latih yang di tersebut dilakukan klasifikasi menggunakan metode *Random Forest Classifier* dan dilakukan pengujian terhadap data testing. Berdasarkan hasil yang dilakukan akurasi tertinggi di dapatkan pada perbandingan data 8:2 dengan akurasi 85%, sedangkan untuk perbandingan data 9:1 di dapat dengan akurasi 81%, data dengan perbandingan 7:3 di dapat dengan akurasi 83%, data dengan perbandingan 6:4 di dapat dengan akurasi 82%, dan data dengan perbandingan 5:5 di dapat dengan akurasi 81%. Dapat di simpulkan bahwa pada penelitian ini data yang digunakan cukup penting hingga dapat mempengaruhi hasil akurasi pada algoritma yang digunakan. Untuk penelitian selanjutnya dibutuhkan data yang lebih banyak untuk menambah tingkat akurasi.

Kata kunci : Klasifikasi, Data Mining, Twitter, Bot, Random Forest Classifier.

PRAKATA

Puji Syukur penulis panjatkan kehadirat Allah SWT atas segala berkah, rahmat, dan karunia-Nya, sehingga penulis dapat menyelesaikan Laporan Skripsi yang berjudul "Klasifikasi Bot Twitter dengan menggunakan Random Forest Classifier" dapat diselesaikan dengan baik.

Pada kesempatan ini, Penulis menyampaikan ucapan terima kasih yang sebesar - besarnya kepada:

1. Saiful Anwar, S. Tp, MP selaku Direktur Politeknik Negeri Jember.
2. Hendra Yufit Riskiawan, S.Kom, M.Cs selaku Ketua Jurusan Teknologi Informasi.
3. Trismayanti Dwi P, S.Kom. M.Cs selaku Ketua Prodi Teknik Informatika Jurusan Teknologi Informasi
4. Seluruh dosen Program Studi Teknik Informatika di Politeknik Negeri Jember yang sudah memberikan ilmu yang berharga bagi penulis dan membantu dalam menyelesaikan skripsi ini.
5. Kedua orang tua yang telah memberikan doa, motivasi dan semangat dalam proses pengerjaan proposal skripsi ini.
6. Teman-teman Teknik Informatika angkatan 2019 yang telah mebantu dan memberikan motivasi selama proses pengerjaan proposal skripsi ini.

Penulis menyadari bahwa proposal skripsi ini masih banyak kekurangan. Oleh karena itu Penulis mengharapkan kritik dan saran yang sifatnya membangun guna perbaikan di masa mendatang. Semoga tulisan ini bermanfaat.

Penulis

DAFTAR ISI

Halaman

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

BAB 1. LATAR BELAKANG

1.1. Latar Belakang

Penggunaan media sosial yang semakin meluas juga tingginya peminat sudah menjadi kebutuhan sehari – hari bagi masyarakat. Peningkatan dalam penggunaan media sosial tidak hanya menjadikan media soasial sebagai alat komunikasi yang populer, tetapi juga menjadi sumber informasi, hiburan, dan wadah untuk berbagi pandangan dan pengalaman. Banyaknya penggunaan media sosial di kehidupan menjadi salah satu faktor pendukung dalam meningkatnya penggunaan media sosial dalam penyebaran informasi. Tingkat respon masyarakat terbukti lebih *responsive* terhadap adanya sebuah kejadian atau sebuah fenomena yang ada pada media sosial, itu menjadikan media sosial sebagai wadah media komunikasi yang cocok untuk masyarakat. Media sosial juga telah menjadi alat yang memberikan kenyamanan bagi masyarakat. Kehadiran media sosial, salah satunya ialah *Twitter* yang telah mengubah sudut pandang masyarakat tentang bagaimana caranya berinteraksi dan berkomunikasi dengan orang lain di seluruh dunia. *Twitter* menjadi media sosial yang sering digunakan untuk menjadi alat komunikasi(Azmi et al., 2021).

Twitter menjadi media sosial yang populer dan banyak di pakai oleh kalangan milenial ataupun dewasa, persentase pengguna *Twitter* yang banyak sekali digunakan dan bahkan masuk dalam peringkat 5 terbesar berdasarkan intensitas penggunaannya dibandingkan dengan media sosial lainnya. Di Indonesia dari tahun ke tahun juga terus mengalami peningkatan yang pesat. Pada tahun 2019, pengguna *Twitter* di Indonesia bahkan mencapai 6,43 juta pengguna atau sekitar 52% dari total pengguna media sosial(Azmi et al., 2021).

Twitter yang merupakan salah satu platform media sosial dengan layanan *micro blogging* yang dapat menyampaikan pesan singkat berupa batasan karakter tidak lebih dari 140 karakter dengan bersifat publik dan dapat dilihat oleh pengguna lain. Pengguna *Twitter* biasnya mengekspresikan diri mereka dengan melakukan *tweet* tentang opini dan membahas isu – isu yang ramai saat ini.

Namun, meskipun kehadiran *Twitter* memberikan banyak manfaat, terdapat juga beberapa risiko dan tantangan yang harus dialami oleh banyak pengguna. Salah satu masalah utama yang dihadapi oleh *Twitter* adalah tingginya tingkat kejahatan yang terkait dengan banyaknya akun bot atau akun palsu pada platform *Twitter*(Aditya et al., 2019).

Permasalahan ini menimbulkan tingginya tingkat kejahatan pada platform *Twitter* karena banyak pihak-pihak yang tidak bertanggung jawab dengan sengaja membuat akun *Twitter* yang palsu untuk membahayakan pengguna lainnya. Kejahatan yang dilakukan ini memiliki peran yang sangat signifikan dalam mengacaukan data dimana sebuah akun bot dapat dapat di operasikan secara otomatis seperti memposting, mengkomentari, atau berinteraksi dengan konten lainnya pada platform *Twitter*(Aditya et al., 2019). Dampak buruk dari keberadaan bot adalah konten *spam* yang memungkinkan bot dapat menyebarluaskan informasi palsu (hoax), konten negatif dan berbahaya secara otomatis kepada pengguna *Twitter*(Ruth et al., 2019).

Bot pada *Twitter* dibuat dengan menggunakan skrip atau program dari *computer*. Akun bot ini biasanya dibuat secara banyak oleh orang yang sebenarnya. Bot ini sering digunakan untuk berbagai tujuan yang merugikan, seperti mempengaruhi opini publik, menyebarkan *spam*, atau bahkan menyebarkan informasi palsu dan *link* berbahaya. Dengan adanya Bot pada *Twitter* dapat mengganggu pengguna di platform *Twitter* yang ingin berinteraksi dengan akun-akun manusia yang asli. Selain banyaknya dampak buruk dengan adanya penggunaan bot, adapun cara konvensional untuk mendeteksi akun bot dan akun nyata, dapat dilakukan dengan mengamati pola aktivitas pada sebuah akun. Misalnya, memperhatikan bahwa akun tertentu melakukan lebih banyak 1 perilaku seperti *retweet* dari pada membuat *tweet* asli, menulis banyak *tweet*, tetapi hanya memiliki beberapa pengikut. Selain itu, akun tersebut juga tidak memiliki biografi, gambar profil, dan menulis konten *tweet* yang sama dengan pengguna lain secara bersamaan. Namun, pendekatan kognitif seperti itu dinilai tidak efisien dan hanya fokus pada presisi(Aditya et al., 2019).

Upaya untuk mendeteksi dan mengatasi akun bot dan akun palsu terus dilakukan oleh *Twitter* dan para peneliti. Metode yang dipakai semakin banyak dan berkembang dengan menggunakan teknik-teknik kecerdasan buatan dan pembelajaran mesin untuk mengidentifikasi pola perilaku yang mencurigakan. Selain itu, kerjasama antara pengguna, peneliti, dan platform media sosial juga penting dalam melawan penyebaran konten merugikan dan melindungi pengguna dari ancaman yang mungkin timbul(Daffa et al., 2018).

Dengan adanya permasalah tersebut yang melibatkan akun bot yang berdampak mengganggu data, dibutuhkan penggunaan *machine learning* agar memungkinkan dalam pengembangan algoritma yang cerdas untuk menganalisis pola dan perilaku setiap akun pada platform *Twitter*. Dengan mempelajari karakteristik akun akun yang terbukti sebagai bot, machine learning dapat menggunakan pola pola tersebut untuk mengidentifikasi kemungkinan keberadaan bot di masa mendatang(Ruth et al., 2019).

Terdapat penelitian sebelumnya yang sudah dilakukan menggunakan algoritma yang sama untuk mendeteksi sebuah akun bot pada platform *Twitter* yang dilakukan oleh Aqila Aini Zahra yang berjudul *"bot detection application on Twitter using maching learning with random forest classifier algorithm"* pada tahun 2020 (Zahra et al., 2020). Dari hasil penelitian tersebut sistem deteksi bot menggunakan *machine learning* menghasilkan label pada *username* yang dicari dengan model akurasi mencapai 96% berdasarkan Analisa *profile* pengguna dan Analisa *tweet* pengguna *Twitter*.

Maka pada penelitian ini digunakan metode yang sama dengan penelitian sebelumnya, yaitu *Random Forest Classifier* untuk mendeteksi sebuah akun dinyatakan bot atau tidak. Sumber data yang akan digunakan pada penelitan ini diambil dari media sosial *Twitter* menggunakan *API* (*application Programming Interface*) yang telah disediakan. Jumlah data yang digunakan pada penelitian ini adalah 1200 data sebagai data latih dan 10 data sebagai data uji. Penentuan jumlah data tersebut didasari dari penelitian sebelumnya yang dilakukan oleh (Zahra et al., 2020). Dengan demikian klasifikasi algoritma *Random Forest* dapat dilakukan lebih akurat. Pada penelitian ini data yang diambil menyangkut data

dari *profile* sebuah akun pada *Twitter* seperti analisa pola *tweet* dan analisa pola *profile.* Hasil analisa perhitungan pola dengan menggunakan *Random Forest* akan di muat pada sebuah sistem *GUI* (*Graphical User Interface*) dengan cara menghasilkan prediksi sebuah label pada data akun dinyatakan bot atau *human*.

1.2. Rumusan Masalah

Berdasarkan paparan pada latar belakang di atas, maka dapat disimpulkan permasalahan yang mendasari dari terlaksananya penelitian ini yaitu Bagaimana menerapkan hasil implementasi metode *Random Forest Classifier* untuk mengetahui sebuah akun bot atau *human* pada platform *Twitter*.

1.3. Tujuan

Berdasarkan rumusan masalah yang telah di paparklan diatas, tujuan dari penelitian ini adalah sebagai berikut :

1. Mengetahui hasil implementasi metode *Random Forest Classifier* untuk mendeteksi sebuah akun *Twitter* dinyatakan bot atau *human*.

1.4. Manfaat

Hasil dari penelitian ini diharapkan dapat memberikan manfaat antara lain sebagai berikut :

1. Bagi Mahasiswa

   penelitian ini diharapkan dapat memberikan manfaat bagi mahasiswa untuk menggunakan penelitian ini sebagai referensi guna membantu proses pengembangan pada penelitian berikutnya.

2. Bagi Masyarakat

   Penelitian ini diharapkan dapat memberikan manfaat bagi Masyarakat, khususnya bagi orang – orang yang ingin membedakan akun bot dan *human*.

BAB 2. TINJAUAN PUSTAKA

2.1 *State of the Art*

Deteksi Bot *Spammer* pada *Twitter* Berbasis *Sentiment Analysis* dan *Time Interval Entropy* (Christian Sri Kusuma Aditya, Mamluatul Hani'ah, Alif Akbar Fitrawan, Agus Zainal Arifin, Diana Purwitasari) Penelitian yang dilakukan dengan mengintegrasikan *Sentiment Analysis* (*SA*) yang berdasarkan emosi dan *Time Interval Entropy* (*TIE*). *Sentiment Analysis* (*SA*) digunakan untuk mendeteksi ungkapan ekspresi ataupun opini yang terkandung dalam *tweet*. Sedangkan *Time Interval Entropy* (*TIE*) digunakan untuk menangkap keteraturan waktu memposting *tweet* yang menunjukkan *tweet* diunggah secara otomatis. Hasil percobaan menunjukan bahwa *precision* dan *recall* dari metode yang diusulkan mencapai 83% dan 91%. Hal ini membuktikan penggabungan *Sentiment Analysis* (*SA*) dan *Time Interval Entropy* (*TIE*) dapat mengoptimalkan performa sistem secara keseluruhan dalam mendeteksi Bot *Spammer*.

Deteksi *Twitter* Bot Menggunakan Klasifikasi *Decission Tree* (Hendra Kurniawan) Penelitian ini bertujuan untuk melakukan pendeteksian akun *Twitter* bot pada media sosial *Twitter* dengan menggunakan klasifikasi *Decission Tree*. Dengan menggunakan metode klasifikasi *Decission Tree* dengan perhitungan *timestamp* berdasarkan *tweet* dan *retweet* yang dilakukan. klasisfikasi yang dilakukan dengan membandingkan pola *tweet* dan *retweet* dengan pola dari kelompok pengguna secara umum. Dengan menggunakan *Decission Tree*, Hasil pengukuran menunjukkan performa *accuracy* model mencapai 88.84% dan perhitunga kurva *UAC* dengan nilai 0.965.

Identifying Fake News *on Twitter using Naive Bayes*,*SVM*, *Random* Forest *Distributed Algorithms* (Adrian Iftene, Ciprian Cusmuliuc) Penelitian yang dilakukan menggunakan Algortima *Random Forest Classifier* dengan membuat sebuah pohon keputusan sebanyak 10 pohon. Data yang diaambil secara acak dan nantinya di pilih pilihan yang terbaik.Hasil akurasi yang didapat adalah 95,93%.

Untuk mengetahui perbedaaan tiap penelitian berdasarkan parameter dan metode yang sudah dilakuakan di atas dapat dilihat secara rinci pada Tabel 2.1.

Tabel 2.1 Penelitan yang sudah dilakukan sebelumya

No	Penulis	Judul	Metode	Parameter
1	Christian Sri Kusuma Aditya, Mamluatul Hani'ah, Alif Akbar Fitrawan, Agus Zainal Arifin, Diana Purwitasari	Deteksi Bot *Spammer* pada *Twitter* Berbasis *Sentiment Analysis* dan *Time Interval Entropy*	*Time Interval entrophy*	Input : *tweet* akun pengguna, *time interval tweet* Output: mendapatkan hasil grafik data negatif dan positif
2	Hendra Kurniawan	Deteksi *Twitter* Bot Menggunakan Klasifikasi *Decission Tree*	*Decission tree*	Input : tweet akun pengguna, *retweet* akun pengguna Output: mendapatkan hasil grafik data akurasi model
3	Adrian Iftene, Ciprian Cusmuliuc	*Identifying Fake News on Twitter using Naive Bayes*,*SVM*, *Random Forest Distributed Algorithms*	*Naïve bayes*, *SVM*, *Random Forest*	Input : *tweet* akun pengguna Output: mendapatkan hasil grafik data akurasi model

2.2 Media Sosial

Media Sosial merupakan platform yg bisa dipakai sang seluruh orang lantaran kemudahannya pada bertukar informasi,membuatkan foto & video menciptakan tulisan, membentuk blog, sampai bisa memperlihatkan fasilitas dalam penggunanya secara bebas buat menaruh pendapat misalnya dalam media umum *Twitter*. penggunaan media sosial umum sangat semakin tinggi & membuahkan media umum terpopuler dikalangan warga terutama para pelajar. para pelajar merupakan pengguna paling aktif yg acapkalikali memakai *Twitter* & memanfaatkannya menjadi media belajar(Rezeki, 2020).

2.3 *Data Mining*

Data mining merupakan suatu teknik yang digunakan untuk menggali atau menguraikan data dengan tujuan untuk mendapatkan informasi yang berharga. Proses *data mining* melibatkan penggunaan berbagai teknik seperti statistik, matematika, kecerdasan buatan, dan pembelajaran mesin *(machine learning)* guna mengekstraksi informasi yang dapat memberikan manfaat dari berbagai *database*.Dalam proses *data mining*, data yang tersedia dianalisis dengan menggunakan berbagai algoritma pembelajaran. Teknik statistik digunakan untuk menganalisis pola dan hubungan dalam data, sedangkan matematika digunakan untuk mengembangkan model dan metode yang tepat dalam pengolahan data. Selain itu, kecerdasan buatan memainkan peran penting dalam menerapkan algoritma yang canggih untuk mengidentifikasi pola yang kompleks dan mendapatkan wawasan yang lebih dalam. Pembelajaran mesin juga menjadi bagian integral dari data mining, di mana algoritma dapat belajar dari data yang ada dan membuat prediksi atau klasifikasi berdasarkan pola yang ditemukan(Utomo, 2020).

2.4 *Data Science*

Data Science merupakan ilmu yang di kembangkan dengan konsep DDDDM (*Data-Driven Decision Making*). *Data Science* menggunakan metode ilmiah, algoritma komputasi, dan sistem informasi untuk menggali wawasan dan

pengetahuan dari data yang ada. *Data Science* menggabungkan prinsip dan teknik dari statistik, matematika, ilmu komputer, dan domain spesifik untuk menganalisis, memahami, dan mengambil keputusan berdasarkan data. Beberapa teknologi yang sangat mendukung *Data Science* termasuk *machine learning* digunakan untk memahami pola, tren, dan informasi berharga dari data yang dapat digunakan untuk mengoptimalkan prediksi, mengidentifikasi peluang, wawasan yang mendalam dalam berbagai bidang(Adhisyanda, 2020).

2.5 *Artificial Intelligence*

Artificial Intelligence atau kecerdasan buatan adalah teknologi yang memungkinkan komputer untuk melakukan tugas-tugas yang sebelumnya hanya bisa dilakukan oleh manusia. Dengan menggunakan berbagai metode seperti *machine learning* dan algoritma cerdas, *Artificial Intelligence* dapat belajar dari data, mengenali pola, dan mengambil keputusan berdasarkan informasi yang diberikan. Proses pembelajaran *Artificial Intelligence* adalah dengan memungkinkan sistem untuk membaca atau mengidentifikasi pola, membuat kesimpulan, dan mengambil keputusan yang cerdas.

Sistem *Artificial Intelligence* belajar dengan mempelajari contoh-contoh yang diberikan dan secara mandiri *Artificial Intelligence* akan menemukan pola dan hubungan dalam data. Dengan adanya pembelajaran, sistem *Artificial Intelligence* dapat meningkatkan kinerjanya seiring waktu dan pengalaman. Selain itu, *Artificial Intelligence* juga menggunakan proses berpikir yang mirip seperti manusia. Sistem *Artificial Intelligence* mampu menganalisis informasi yang ada.(Sobron & Lubis, 2021).

2.6 *Twitter*

Twitter adalah layanan sosial media berkarakteristik mikroblog yg memiliki minat paling banyak di kalangan milenial sampai dewasa. Twitter juga membuat pengguna mengirimkan/menulis teks sampai 280 karakter. dari tahun didirikannya Twitter dalam bulan maret 2006(Zahra et al., 2020). kepopuleran Twitter didukung dengan makin banyaknya pengguna setiap harinya.ini juga membuat Twitter masuk peringkat dua menjadi media umum yg acapkalikali dikunjungi

pada global. Kepercayaaan & minat pada kalangan warga menciptakan layanan Twitter dalam tahun Indonesia bertambah 10 Juta dari tahun 2020, dan Twitter menduduki peringkat 5 teratas media sosial yang paling sering digunakan di Indonesia.(Wandani et al., 2021) Pada media sosial Twitter kita bisa berkomentar atau pesan yg bisa dibaca sang Twitter User lainnya, fitur ini biasa kita sebut tweet atau kicauan. Pada Twitter kita bisa menciptakan tweet menggunakan maksimum 140 karakter yg menciptakan media umum tadi unik Sebagai asal informasi, Komponen Twitter yang dapat digunakan untuk pengambilan informasi sepeti Username,*Hastag*,waktu tweet dikirim, *reply*, dan *retweet*.

2.7 Twitter API *(Application Programming Interface)*

Twitter API (*Application Programming Interface*) adalah cara buat menghubungkan antar personal komputer supaya bisa meminta & menampilkan sebuah kabar. teknik yang dilakukan untuk menghubungkan antar aplikasi agar dapat menyampaikan dan menampilkan sebuah informasi. dengan *Twitter API* (*Application Programming Interface*) membuat pengguna dapat mengambil dan menampilkan data yang disediakan oleh Twitter.(Adi Yahyadi, 2022). Sumber API (*Application Programming Interface*) yang telah disediakan oleh platform *Twitter* dapat dilihat secara rinci pada Lampiran 3.

2.8 Bot *Twitter*

Sebuah akun yang menghasilkan konten otomatis di latar belakang *Twitter* dan pasti dapat mengikuti dan menarik perhatian pengguna *Twitter*. Penjahat dunia maya mencoba mengontrol bot ini untuk mengontrol akun pengguna untuk mengancam privasi dan keamanan mereka. Bot biasanya dibuat secara otomatis dan dapat juga melakukan tugas secara otomatis(Daffa et al., 2018).

2.9 *Machine Learning*

Machine Machine learning merupakan sistem yang mampu belajar sendiri untuk memutuskan sesuatu tanpa harus berulangkali diprogram oleh manusia

sehingga komputer menjadi semakin cerdas berlajar dari pengalaman data yang dimiliki.(Retnoningsih & Pramudita, 2020).

2.10 Klasifikasi

Klasifikasi merupakan metode yang paling umun digunakan pada *data mining* ini dilakukan tindakan pengelompokan pada setiap keadaan.Setiap keadaan yang akan berisikan sekelompok atribut yang disebut *class atribute*. Klasifikasi membantu menemukan label kelas yang sesuai untuk digunakan pada *supervised learning* yang membutuhkan kumpulan data yang diberi label dengan baik untuk *training* data(Utomo, 2020).

2.11 *Random Forest Classifer*

Random Forest merupakan metode atau algoritma dari teknik pohon keputusan yang digunakan untuk pengklasifikasi dan regresi. Metode ini merupakan sebuah *esemble* (kumpulan) pohon keputusan sebagai dasar dari *Random Forest Classifier* yang dibangun dan dikombinasikan. Salah satu aspek penting dalam metode *Random Forest* adalah penggunaan *bootstrap sampling* untuk membangun pohon prediksi. Setiap pohon keputusan dalam *Random Forest* memprediksi secara acak, dan *Random Forest* itu sendiri melakukan prediksi keputusan berdasarkan hasil *voting* dari seluruh pohon dalam *ensamble*.

Model dari *Random Forest* terdiri dari *K* pohon keputusan. Setiap pohon memilih variabel *x* yang paling sesuai dengan pohon tersebut. Dalam proses klasifikasi, suara tunggal diberikan kepada pohon yang dipilih, dan prediksi akhir dilakukan berdasarkan mayoritas suara dari pohon-pohon tersebut. Hal ini memungkinkan *Random Forest* dapat mengatasi masalah *overfitting* dan memberikan hasil prediksi yang lebih stabil dan akurat.

Random Forest Classifier juga memiliki kemampuan untuk melakukan pengklasifikasian dengan variabel target yang memiliki kelas diskrit serta regresi dengan variabel target yang bersifat berkelanjutan. Metode ini sangat populer dan efektif dalam banyak aplikasi, termasuk dalam analisis data, pemrosesan citra, dan

bioinformatika. Kelebihan lain dari *Random Forest* adalah kemampuannya dalam menangani data yang tidak seimbang dan kemampuan untuk mengidentifikasi pentingnya setiap variabel dalam klasifikasi. Dengan kombinasi dari pohon keputusan acak, *Random Forest* mampu memberikan prediksi yang handal dan dapat diandalkan. Adapun tahapan yang dilakukan random forest sebagai berikut :

a. *Bagging*

diakukan pemilihan sampel acak dari data latih dengan penggantian. Ini berarti setiap sampel dapat dipilih lebih dari sekali, dan beberapa sampel dapat diabaikan. ini memungkinkan setiap pohon dalam *Random Forest* memiliki data pelatihan yang berbeda-beda. Dengan menggabungkan berbagai sampel acak menghasilkan keragaman di antara pohon-pohon dalam hutan, yang mendukung pembentukan model yang lebih kuat(Rahmi et al., 2023).

b. Pertumbuhan pohon

Proses pertumbuhan pohon dimulai dengan memilih variabel acak sebagai pembagi yang menghasilkan pemisahan terbaik antara kelas dalam data pelatihan. Dalam setiap langkah, data pelatihan dibagi berdasarkan nilai pembagi yang dipilih, membentuk cabang-cabang baru dalam pohon(Rahmi et al., 2023).

c. Prediksi

Setelah semua pohon tumbuh, *Random Forest* digunakan untuk membuat prediksi. Setiap pohon dalam hutan memberikan prediksi berdasarkan fitur *input*. Jika menggunakan *Random Forest* untuk klasifikasi, prediksi akhir dihasilkan dengan memilih kelas yang paling sering muncul di antara prediksi pohon-pohon tersebut. Jika kita menggunakan *Random Forest* untuk regresi, prediksi akhir dihasilkan dengan mengambil rata-rata dari prediksi pohon-pohon tersebut. Dengan menggabungkan hasil prediksi dari pohon-pohon yang berbeda, *Random Forest* dapat menghasilkan prediksi yang lebih stabil dan akurat daripada menggunakan satu pohon saja(Jatmiko et al., 2019).

Entropy dibuthkan dalam metode *Random Forest* untuk mengukur tingkat ketidakmurnian atau ketidakteraturan suatu himpunan data. Dalam konteks *Random Forest*, *entropy* digunakan untuk menentukan atribut mana yang paling baik dalam memisahkan data menjadi kelompok yang homogen. Semakin rendah nilai *entropy*, semakin baik atribut tersebut dalam melakukan pemisahan. Dengan mempertimbangkan *entropy*, *Random Forest* dapat memilih atribut yang paling informatif dan efektif dalam membangun pohon keputusan, sehingga meningkatkan akurasi dan kualitas prediksi dari model(Sandag, 2020).

$$Entropy\ (Y)\ =\ -\sum ip(c|Y)log2p(c|Y)\ \text{.....2.1}$$

Keterangan :
Y = Himpunan kasus
$P(c|Y)$ = Proporsi nilai Y terhadap kelas c.

$$Entropy\ (Y)-\ \sum v\varepsilon values(a)\frac{Yv}{Ya}Entropy\ (Yv).\ \text{.....2.2}$$

Keterangan :
Values (a) = Nilai yang mungkin dalam himpunan kasus a.
Yv = Subkelas dari Y dengan kelas v yang berhubungan kelas a.
Ya = Semua nilai yang sesuai dengan a.

2.12 Pengujian

Pengujian dilakukan terhadap algoritma *Random Forest* dan terhadap sistem. Algoritma *Random Forest* diuji menggunakan metode akurasi dan klasifikasi pada *Confusion Matrix*.

2.12.1 *Confusion Matrix*

Confusion Matrix merupakan suatu metode yang digunakan untuk melakukan perhitungan akurasi pada proses klasifikasi. Sehingga hasil evaluasi dengan *Confusion Matrix* berupa nilai *accuracy*, *presisi* dan *recall*. *Presisi* dan *recall* merupakan istilah yang akan muncul apabila sistem yang sudah dibuat dapat menampilkan hasil (*retrieve*) suatu hasil baik berupa klasifikasi, prediksi dan pencarian(Hary Candana et al., 2021).

Tabel 2.2 *Confusion Matrix*

		Nilai Sebenarnya	
		True	False
Nilai Prediksi	True	True Positive(TP)	False Positive(FP)
	False	False Negative	True Negative

$$mean\ of\ accuration \sum_{i=1}^{k} \frac{tp_i+tp_i}{tp_i+tn_i+fp_i+fn_i} \quad \text{.....2.3}$$

$$precission_{\mu} = \frac{\sum_{i=1}^{k} tp_i}{\sum_{i=1}^{k}(tp_i+fp_i)} \text{.....2.4}$$

$$recall = \frac{\sum_{i=1}^{k} tp_i}{\sum_{i=1}^{k}(tp_i+fn_1)} \text{.....2.5}$$

Keterangan :
Mean of accuration = rasio prediksi Benar dengan keseluruhan data
Precission = rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positf
Recall = rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif

2.13 *Python*

Python merupakan salah satu Bahasa pemograman yang direkomendasikan untuk digunakan pada *machine learning*. Dalam beberapa tahun terakhir, Dukungan *library Python* yang ditingkatkan telah membuatnya menjadi alternatif yang kuat untuk tugas analisis data(Sodik et al., 2020).

2.14 *Flowchart*

Flowchart adalah prosedur atau program yeng memiliki alur (*flow*) berupa sistem secara logika.untuk merepresentasikan simbol- simbol tertentu yang mudah dipahami, dan mudah digunakan.tujuan dari *Flowchart* adalah menggambarkan masalah secara sederhana(Syamsiah, 2019).

BAB 3. METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Tempat proses pelaksanaan penelitian ini dengan judul "Klasifikasi Bot *Twitter* menggunakan *Random Forest Classifier*" dengan tempat penelitian di Politeknik Negeri Jember, dengan waktu pelaksanaan 4 bulan.

3.2 Alat dan Bahan

3.2.1 Alat

Alat – alat yang digunakan dalam penelitian ini yaitu perangkat keras dan perangkat lunak sebagai berikut :

a) Perangkat keras

1. Laptop : Acer E5-476G
2. RAM : 12GB
3. Processor : Intel Core I3-8130U
4. Handphone : Poco M3
5. Penyimpanan : 1 TB

b) Perangkat lunak

1. Sistem Operasi : *Windows 11, VPS*
2. Bahasa Pemograman : *Python*
3. Software : *Jupyter Notebook*

3.2.2 Bahan

Bahan yang digunakan dalam penelitian ini untuk menjadi pertimbangan sebuah akun dinyatakan bot atau bukan pada platform *Twitter* menyangkut data yang diambil sebanyak 1200 data sebagai data latih dan 10 data sebagai data uji yang dapat dilihat pada Tabel 3.1.

Tabel 3.1 Bahan

Jenis Analisis	Parameter	Jenis Paramter
Analisis Profile	Friends_count	Dasar
	Default_profile	Dasar
	Verified	Dasar
	Followers_count	Dasar
	Statuses_count	Dasar
	Default_profile	Dasar
	Screen_name	Dasar
	Favourites_count	Dasar
	Location	Dasar
	Created_at	Dasar
	Account_age_days	Dasar
	Contains_bot_name	Turunan
	Ratio_statuses_count_per_age	Turunan
	Ratio_favourites_per_age	Turunan
	Ratio_friends_per_followers	Turunan
	Reputation	Turunan
Analisis konten	Description	Dasar
	Avg_word	Turunan
	Word_count	Turunan
	Char_count	Turunan

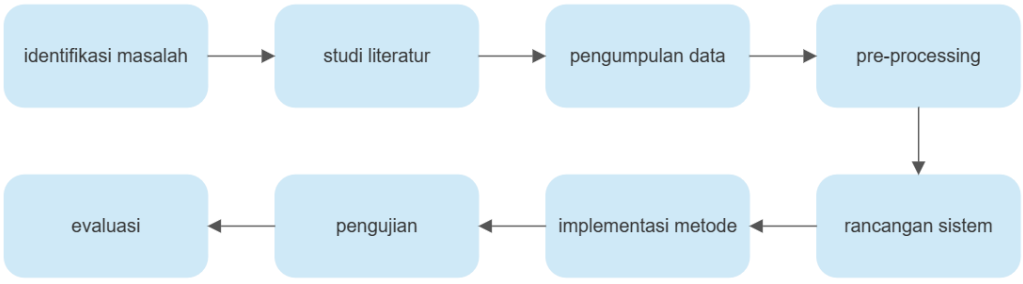
Semua parameter ini adalah indikator potensial yang digunakan untuk menentukan apakah sebuah akun *Twitter* adalah bot atau *human*, dengan menggunakan metode *Random Forest* sebagai alat untuk menggabungkan dan menganalisis informasi tersebut secara akurat. Terdapat jenis parameter yang mewakili nama parameter tersebut seperti parameter dasar dan parameter turunan.

Parameter dasar adalah jenis *feature* yang tersedia secara langsung pada data yang dikumpulkan dari *API* (*Application Programming Interface*), seperti

jumlah kata, karakter, atau *retweet* pada setiap *tweet*. Sementara itu, parameter turunan adalah jenis *feature* yang memerlukan proses kalkulasi tambahan, seperti melakukan perhitungan pada parameter dasar untuk memperoleh nilai dari parameter tersebut. Ini mengacu pada variabel atau atribut yang diperoleh atau dihasilkan dari data *Twitter*, maka parameter turunan digunakan untuk merujuk pada fitur-fitur atau metrik yang dihitung dari parameter dasar pengguna *Twitter*(Seno & Wibowo, 2019).

3.3 Tahap Penelitian

Dalam penelitian ini, diperlukan rancangan urutan kegiatan penelitian yang akan dilakukan oleh peneliti yang mencakup langkah-langkah penelitian sebagai panduan dalam melaksanakan penelitian. Proses penelitian dapat diilustrasikan melalui diagram dan dapat dilihat pada Gambar 3.1.



Gambar 3.1 Tahapan Penelitian

3.3.1 Identifikasi Masalah

Identifikasi masalah diperhatikan seperti data, pemilihan metode yang tepat, dan interpretasi yang hati-hati, mempertimbangkan kelebihan dan kelemahan metode, serta mengambil kesimpulan yang didukung bukti yang kuat dapat meningkatkan kualitas penelitian.

3.3.2 Studi literatur

Studi literatur adalah proses menyelidiki, menganalisis, dan mengevaluasi karya-karya tulis yang relevan dengan topik penelitian. Diperlukan identifikasi dan analisa teori serta temuan terkait sebelumnya. Studi literatur membantu merumuskan pertanyaan penelitian, merancang metodologi, dan menyediakan konteks yang relevan untuk penelitian yang akan dilakukan.

3.3.3 Pengumpulan Data

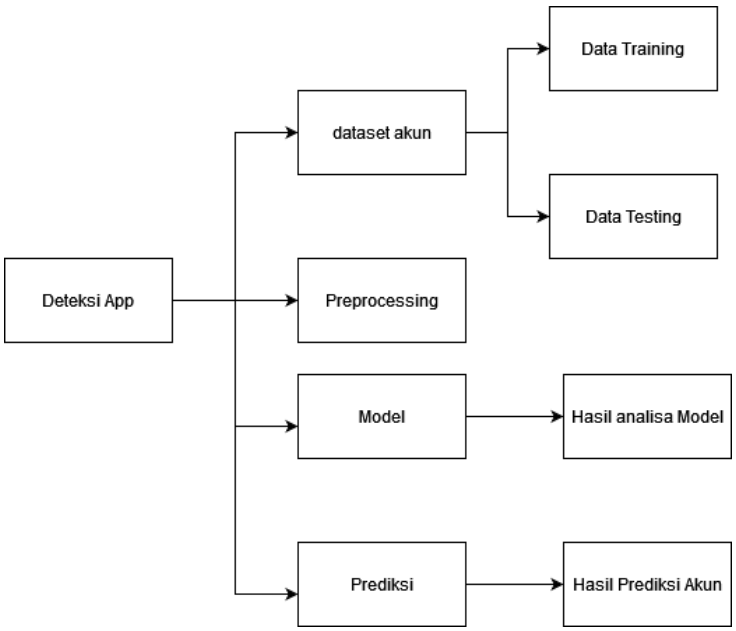
Dalam pengumpulan data yang di ambil dari *Twitter*, sebanyak 1200 data *user profile* sebagai sampel. Data yang diambil berdasarkan dari *source API (Application Programming Interface)* yang dapat dilihat pada Lampiran 3. Data yang sudah di dapat kemudian di *split* (pemisahan data) menjadi data latih dan data uji. Perbandingan data yang digunakan adalah 8:2 dimana data sebanyak 960 menjadi pelatihan (*training*) dan data sebanyak 240 untuk pengujian (*testing*). Pemisahan data tersebut ini penting bertujuan untuk membangun dan mengevaluasi model atau algoritma yang akan digunakan dalam analisis data *Twitter*. Data *training* digunakan untuk melatih model dan mempelajari pola serta fitur dari data, sementara data *testing* digunakan untuk menguji performa model yang telah dilatih dan mengukur keakuratannya.

3.3.4 *Preprocessing*

Preprocessing data merupakan tahap penting dalam analisis data *Twitter*. Pada tahap ini dilakukan *cleaning*(membersihkan) dan mempersiapkan data sebelum diolah lebih lanjut. Proses *preprocessing* melibatkan langkah-langkah seperti mengubah nilai menjadi numerik, mendapatkan parameter turunan, dan menghapus data duplikati. *Preprocessing* data bertujuan untuk mengurangi *noise* atau gangguan dalam data, menghasilkan representasi yang lebih seragam, dan meningkatkan efektivitas analisis yang akan dilakukan. Dengan melakukan Tahap *preprocessing* dapat memperoleh data yang lebih terstruktur dan siap digunakan dalam tahap selanjutnya dalam analisis data *Twitter*.

3.3.5 Rancangan Sistem

Rancangan sistem melibatkan strukturisasi keseluruhan proses analisis. Sistem ini dirancang untuk mengintegrasikan berbagai komponen, seperti pengumpulan data, *preprocessing*, analisis, dan visualisasi hasil. Rancangan sistem juga mencakup pemilihan algoritma dan teknik analisis yang sesuai dengan tujuan penelitian atau analisis yang dilakukan. Dengan ini diperlukan sebuah Rancangan sistem dengan *User Interface (UI)* untuk menampilkan informasi akun yang akan di analisis. Berikut adalah rancangan sistem yang dapat dilihat pada Gambar 3.2.

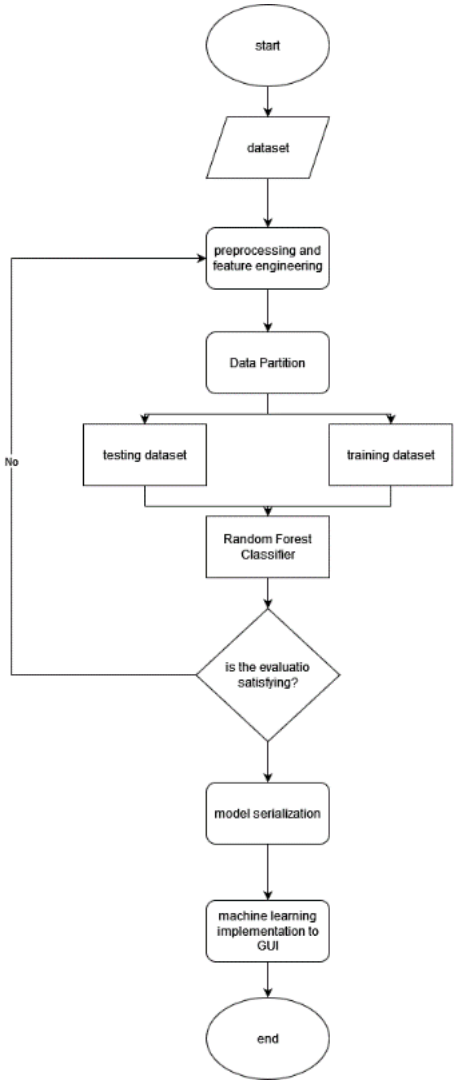


Gambar 3.2 Rancangan Sistem

3.3.6 Implementasi Metode

Dalam penelitian ini melibatkan penerapan langkah-langkah dan Teknik pemecahan masalah yang dirancang dalam rancangan sistem. Pada tahap ini, algoritma *Random Forest* dipilih dan diterapkan pada data yang telah diolah sebelumnya. Implementasi metode juga melibatkan pengaturan parameter dan konfigurasi yang sesuai dengan kebutuhan analisis. Dengan implementasi metode

yang baik, akan dapat menghasilkan keberhasilan analisa data yang bagus. alur metode *Random Forest* dapat dilihat pada Gambar 3.3.



Gambar 3.3 *Flowchart Random Forest*

Implementasi algoritma dilakukan dengan pada data latih dan mengujinya dengan data uji. Implementasi algoritma ini digunakan untuk pengambilan keputusan pada data jika data sesuai maka akan melakukan keputusan selanjutnya hingga semua terpenuhi. Algoritma ini dapat diakses di *scikit-learn* menggunakan *Python* versi 3.9.

Setelah melakukan pelatihan awal, lakukan penyetelan parameter pada model *Random Forest*. Penyetelan parameter dilakukan untuk mencari

konfigurasi parameter yang optimal agar model dapat memberikan performa yang lebih baik. Parameter yang dapat disetel termasuk jumlah pohon keputusan dalam *Random Forest*, jumlah fitur yang dipertimbangkan dalam setiap pemilihan fitur, atau tingkat keseimbangan antara pohon yang dibangun.

3.3.7 *Pengujian UAT (Blackbox Testing)*

Pengujian User Acceptance Testing (UAT) merupakan pengujian yang dilakukan pada berbagai fitur baru di dalam aplikasi yang belum diluncurkan. Dengan melakukan pengujian ini pengembang dapat memahami rancangan yang dibuat sudah sesuai dan memenuhi harapan pengguna. Pada penelitian ini pengujian *UAT* dilakukan dengan menggunakan *blackbox testing*. *Blackbox testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari sebuah perangkat lunak, tester atau penguji dapat mendefinisikan kumpulan kondisi input dan melakukan pengujian pada spesifikasi fungsional program. (Yusmita et al., 2020)

3.3.8 Evaluasi Model

Evaluasi model dilakukan untuk untuk mengukur kinerja dan efektivitas model yang dibangun menggunakan algoritma *Random Forest* dalam analisis data. Evaluasi dilakukan dengan menggunakan berbagai metrik evaluasi seperti pada *confusion matrix* untuk mendapatkan *accuracy*, *presisi*, *recall*, dan *F1-score*(Yusmita et al., 2020).

BAB 4. HASIL DAN PEMBAHASAN

4.1 Identifikasi Masalah

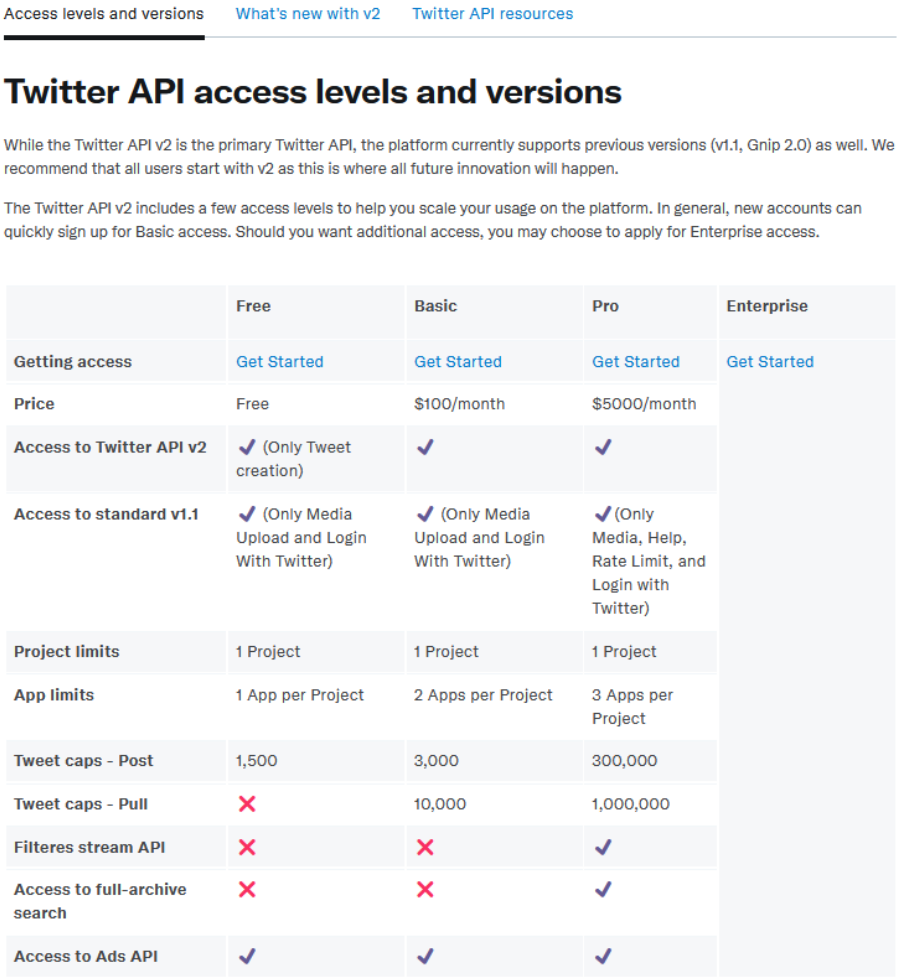
Penelitian dimulai dengan tahap Identifikasi Masalah yang merupakan langkah awal dalam proses penelitian. Pada tahapan ini adalah mengidentifikasi permasalahan yang berkaitan dengan bot *Twitter*. Dilakukan analisis mendalam untuk mengenali dan memahami masalah yang terkait dengan pengenalan dan pemisahan akun bot dan akun *human* di platform *Twitter*.

4.2 Pengumpulan Data

Pada tahap pengumpulan data dilakukan dengan cara *crowling data* (mengambil) data dari *Twitter*. Data yang diambil dapat dilihat pada Tabel 3.1 menliputi parameter parameter dasar seperti *username*, *tweet*, atau *retweet*. Dalam konteks ini, data yang digunakan merupakan *tweet* mentah yang berasal dari *Twitter*. Untuk menjalankan proses pengambilan data tersebut, peneliti memilih menggunakan *code editor Jupyter Notebook* sebagai *Integrated Development Environment* (*IDE*) utama. *Jupyter Notebook* juga dapat lebih mudah melakukan data *cleaning* yang diperlukan sebelum melanjutkan proses Analisa lebih lanjut. Dalam konteks pengambilan *tweet* mentah, data *cleaning* menjadi tahap untuk menghilangkan *noise* atau informasi yang tidak relevan sehingga memperoleh data yang lebih bersih dan berkualitas. *Jupyter Notebook* digunakan untuk menjalankan kode secara langsung dengan mempermudah untuk melihat visualisasi data yang menarik, menyusun teks naratif yang terstruktur, melakukan analisis numerik dan statistik yang mendalam, serta melakukan proses pengolahan data yang efisien.

Sebelum dapat mengakses *Twitter API* menggunakan *library Tweepy*, langkah autentikasi perlu dilakukan dengan memperoleh *API key* melalui permohonan akun pengembang (*Developer Account*) dari alamat *https://developer.Twitter.com/en/docs/Twitter-api.* Setelah permohonan disetujui, pengembang dapat mengakses dan menggunakan *API key* tersebut. Terdapat opsi

pilihan *version* pada halaman *API Key Twitter* dimana ada *API v1* dan *API v2*. Dalam hal ini, peneliti menggunakan jenis *API v1*. *API v1* digunakan karena pilihan ideal bagi pengembang pemula yang melakukan pengujian konsep karena dapat diakses secara gratis.



Access levels and versions | What's new with v2 | Twitter API resources

Twitter API access levels and versions

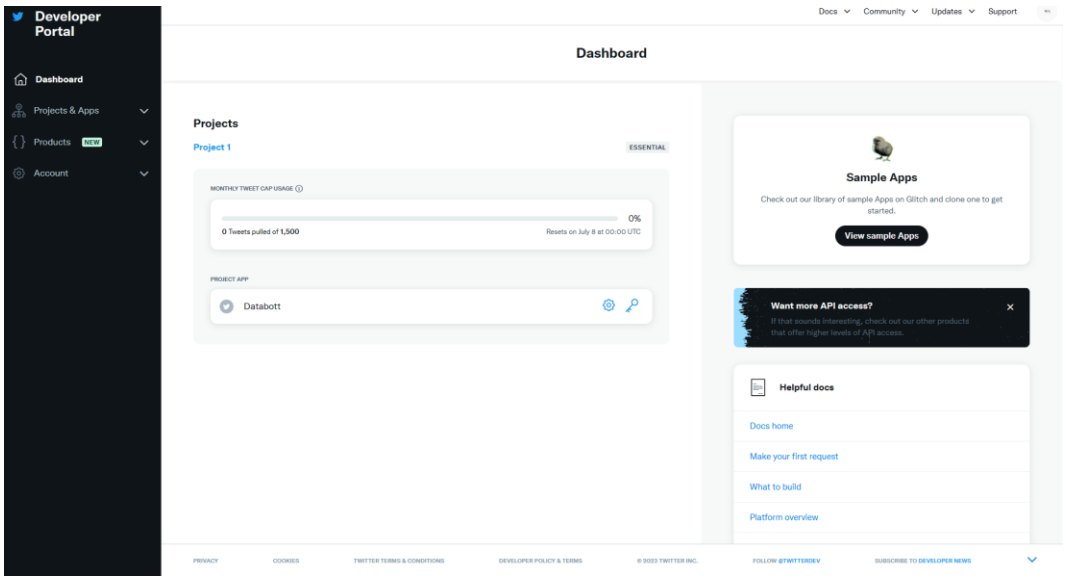
While the Twitter API v2 is the primary Twitter API, the platform currently supports previous versions (v1.1, Gnip 2.0) as well. We recommend that all users start with v2 as this is where all future innovation will happen.

The Twitter API v2 includes a few access levels to help you scale your usage on the platform. In general, new accounts can quickly sign up for Basic access. Should you want additional access, you may choose to apply for Enterprise access.

	Free	Basic	Pro	Enterprise
Getting access	Get Started	Get Started	Get Started	Get Started
Price	Free	$100/month	$5000/month	
Access to Twitter API v2	✓ (Only Tweet creation)	✓	✓	
Access to standard v1.1	✓ (Only Media Upload and Login With Twitter)	✓ (Only Media Upload and Login With Twitter)	✓ (Only Media, Help, Rate Limit, and Login with Twitter)	
Project limits	1 Project	1 Project	1 Project	
App limits	1 App per Project	2 Apps per Project	3 Apps per Project	
Tweet caps - Post	1,500	3,000	300,000	
Tweet caps - Pull	✗	10,000	1,000,000	
Filteres stream API	✗	✗	✓	
Access to full-archive search	✗	✗	✓	
Access to Ads API	✓	✓	✓	

Gambar 4.1 *Twitter API version*

Dalam konteks ini dibutuhkan cukup hanya dengan menggunakan *API key v1* yang dapat dilihat pada Gambar 4.1 karena data yang di berikan sudah cukup untuk menjadi bahan analisa.selanjutnya dengan membuat 1 *project API* dan hanya dapat mengambil 1,500 data dari *Twitter*. Berbeda dengan pilihan lain yang dapat menggunakan lebih dari 1 *project API*. Langkah-langkah autentikasi ini digunakan untuk menghubungkan *API key* dengan *library tweepy*.



Gambar 4.2 *Dashboard API key Twitter*

Pada Gambar 4.2 terdapat 1 *project* dengan nama *Databot* yang digunakan peneliti untuk mengambil data dari *Twitter* melalui *API key* yang sudah di berikan. Dengan demikian, langkah autentikasi ini menjadi langkah awal untuk memulai *crowling data* dari *Twitter*. Data yang di dapat kemudian dijadikan delam bentuk *csv* yang disebut *dataset*. Berikut adalah contoh *dataset* yang telah didapat dari *Twitter* dengan menggunakan *crowling API* yang dapat dilihat pada Tabel 4.1.

Tabel 4.1 Data yang di dapat dari crowling

Nama kolom	Tipe Data	Contoh Data
Screen_name	String	Iyanya
Default_profile	Boolean	False
Description	String	Description 118 #Fever Audio/Visuals OUT on ALL Digital platforms. Find Link Below 👇🏾
Created_at	Datetime	2010-02-26 15:50:31
Favourites_count	String	5130
Followers_count	String	1201365

Friends_count	String	9523
Lanjutan tabel 4.1 Data yang di dapat dari crowling		
Location	String	worldwide
Statuses_count	String	44431
Verified	Boolean	True
Account_age_days	String	3826
Account_type	String	human

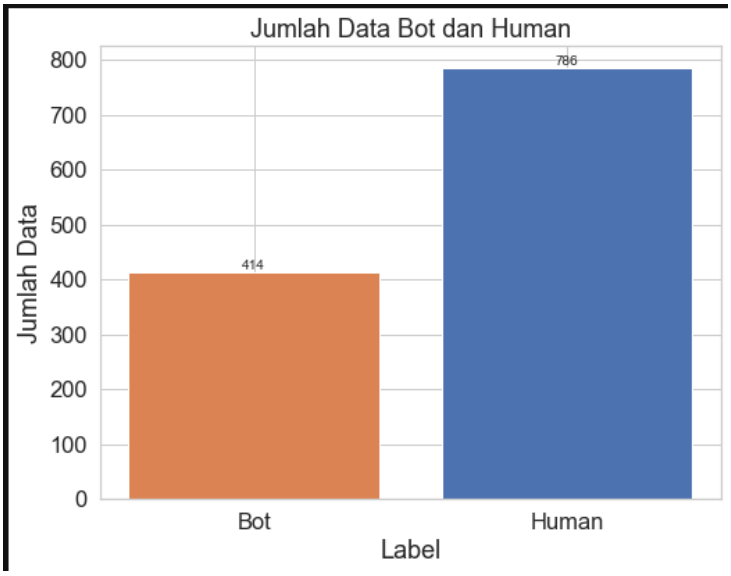
4.3 *Preprocessing*

Setelah data didapat dari crowling maka selanjutnya dilakukan proses pelatihan *machine learning preprocessing* dari dataset. Tujuan *preprocessing* adalah untuk mengurangi kinerja komputasi yang dilakukan mesin, mempercepat waktu, memudahkan pembacaan data, dan meningkatkan akurasi. Data sebanyak 1200 kemudian di *import* menggunakan Bahasa *python* yang dapat dilihat pada Lampiran 4. Hasil dari data yang di *import* dapat dilihat pada Gambar 4.3.

	created_at	default_profile	description	favourites_count	followers_count	friends_count	id	location	screen_name	statuses_count	verified	account_age_days	account_type
0	2014-06-10 13:48:21	True	Cosplay master. Unicorn. Waiting you at https:...	491	0	0	2559057222	unknown	lavenderr90	116	False	2261	bot
1	2010-03-03 05:34:58	True	NaN	1390	16	22	119284832	unknown	sherrik2326384	2033	False	3822	bot
2	2009-01-28 17:30:10	True	The official Twitter account for San Antonio C...	1444	8789	721	19663518	San Antonio, TX	SAC_PR	6026	False	4220	human
3	2009-09-09 15:32:48	False	The Panda's Friend	6013	649266	898	72878707	Los Angeles, CA	MettaWorld37	7224	False	3996	human
4	2009-03-10 14:16:09	False	I've wrestled for the #WWE, I'm an #ECW Origin...	66356	62501	3390	23595924	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human

Gambar 4.3 *preprocessing data*

Berdasarkan dataset tersebut dapat dilihat perbanding awal dengan cara melakukan visualisasi dataset per kategori dengan menggunakan *matplotlib* dan kode dapat dilihat pada Lampiran 5, diketahui bahwa jumlah data untuk kelas *human* dan bot berbeda. Dapat dilihat pada Gambar 4.4.



Gambar 4.4 jumlah data mentah per kelas

Pada visualisasi data diatas jumlah data yang terlabeli diawal sebagai kelas *human* dengan warna biru sebanyak 786 dan kelas bot dengan warna *orange* sebanyak 414 dapat dilihat bahwa kelas bot lebih cenderung lebih sedikit dibandingkan kelas human. Ini disebabkan karena pola dari 1200 akun *profile*, pola data *profile* yang kosong lebih sedikit yang menyebabkan suatu akun dilabeli sebagai *human* lebih banyak dibandingkan dengan bot. setelah dilakukan pelabelan dan visualisasi data terdapat beberapa tahapan yang dilakukan antara lain:

4.3.1 *Data cleansing*

Data cleansing, atau pembersihan data dilakukan untuk menghapus kesalahan atau inkonsistensi dalam data untuk memastikan data tersebut akurat, konsisten, dan dapat dipercaya. Pada data cleansing dilakukan penghapusan duplikat data. Untuk kode selengkapnya dapat dilihat pada Lampiran 6. Data yang sudah di *cleansing* dapat dilihat pada Gambar 4.5.

	created_at	default_profile	description	favourites_count	followers_count	friends_count	id	location	screen_name	statuses_count	verified	account_age_days	account_type
0	2014-06-10 13:48:21	True	Cosplay master. Unicorn. Waiting you at https:...	491	0	0	2559057222	unknown	lavenderr90	116	False	2261	bot
1	2010-03-03 05:34:58	True	NaN	1390	16	22	119284832	unknown	sherrik2326384	2033	False	3822	bot
2	2009-01-28 17:30:10	True	The official Twitter account for San Antonio C...	1444	8789	721	19663518	San Antonio, TX	SAC_PR	6026	False	4220	human
3	2009-09-09 15:32:48	False	The Panda's Friend	6013	649266	898	72878707	Los Angeles, CA	MettaWorld37	7224	False	3996	human
4	2009-03-10 14:16:09	False	I've wrestled for the #WWE, I'm an #ECW Origin...	66356	62501	3390	23595924	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human

Gambar 4.5 Hasil *Data cleansing*

Pada hasil data yang digunakan peneliti tidak terdapat duplikasi seperti nilai dalam sebuah baris sama dengan nilai dalam baris lainnya yang berarti data yang diambil sepenuhnya berbeda atau unik.

4.3.2 *Feature Engineering*

Feature engineering merupakan tahapan seleksi data dalam proses klasifikas. Dalam konteks ini, terdapat dua jenis feature yang digunakan, yaitu parameter dasar dan parameter turunan.

Dalam penelitian ini parameter turunan yang didapat dari parameter dasar meliputi jumlah *retweet*, jumlah *like*, *rasio follower-to-following*, jumlah kata dalam *tweet*, *sentimen tweet*, popularitas pengguna berdasarkan interaksi, atau fitur-fitur lainnya yang dapat memberikan informasi tambahan tentang *tweet* atau pengguna *Twitter*.feature engineering digunakan dalam penelitian ini dalam mengenai pola, tren, atau karakteristik analisa yang relevan untuk tujuan klasifikasi. Proses *feature engineering* ini membantu agar data siap digunakan dalam proses klasifikasi selanjutnya, dengan memberikan informasi dasar dan tambahan yang berharga dalam analisis dan pemodelan data. Data yang digunakan dapat dilihat pada Tabel 4.2.

Tabel 4.2 parameter yang digunakan untuk Analisa

Jenis Analisis	Nama Parameter	Jenis Paramter
Analisis Profile	Friends_count	Dasar
	Verified	Dasar
	Followers_count	Dasar

Lanjutan tabel 4.2 parameter yang digunakan untuk Analisa

	Statuses_count	Dasar
	Description	Dasar
	Default_profile	Dasar
	Favourites_count	Dasar
	Location	Dasar
	Contains_bot_name	Turunan
	Account_age_days	Dasar
	Ratio_statuses_count_per_age	Turunan
	Ratio_favourites_per_age	Turunan
	Ratio_friends_per_followers	Turunan
	Reputation	Turunan
Analisis konten	Avg_word	Turunan
	Word_count	Turunan
	Char_count	Turunan

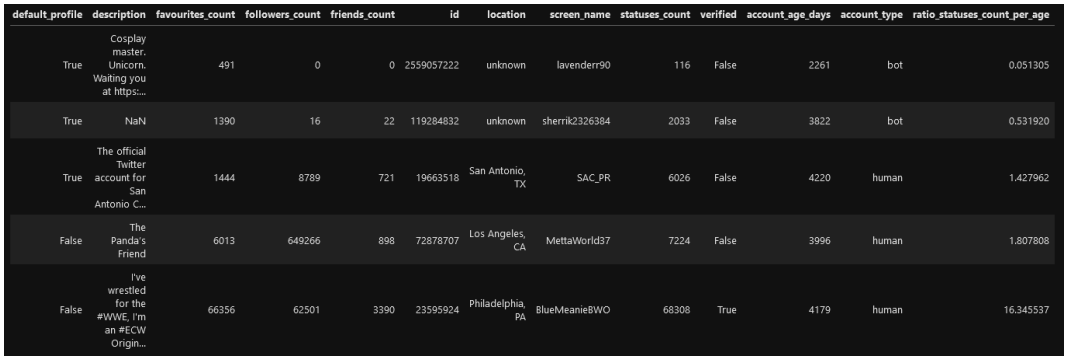
Setelah menentukan daftar *feature* yang akan digunakan dalam proses *training* algoritma, selanjutnya adalah proses perhitungan untuk mendapatkan parameter turunan dari feature tersebut. Terdapat beberapa parameter dasar yang dapat diambil secara langsung saat pengumpulan data. Parameter-parameter tersebut antara lain:

a. *Friends_count* adalah jumlah teman atau pengguna lain yang diikuti oleh pengguna.
b. *Verified* menunjukkan apakah pengguna telah diverifikasi oleh platform.
c. *Followers_count* adalah jumlah pengikut yang dimiliki oleh pengguna.
d. *Statuses_count* adalah jumlah tweet yang telah diposting oleh pengguna.
e. *Default_profile_image* menunjukkan apakah pengguna menggunakan gambar profil default atau bukan.
f. *Default_profile* mengindikasikan apakah pengguna menggunakan profil default atau telah mempersonalisasinya.

g. *Favorites_count* adalah jumlah twit yang telah ditandai sebagai favorit oleh pengguna.
h. *Location* mengacu pada lokasi geografis yang diberikan oleh pengguna pada profilnya.

Dengan mengumpulkan data berdasarkan nilai-nilai parameter dasar ini dapat diambil langsung sebagai *feature* yang akan digunakan dalam proses *pre processing data*. Parameter yang didapat akan langsung ditambahkan ke *dataset* yang digunakan. Hal ini memberikan informasi data yang relevan dan bagus untuk analisis dan pemodelan data selanjutnya. Adapun parameter parameter tambahan yang menjadikannya sebagai parameter turunan untuk meningkatkan performa model sebagai berikut:

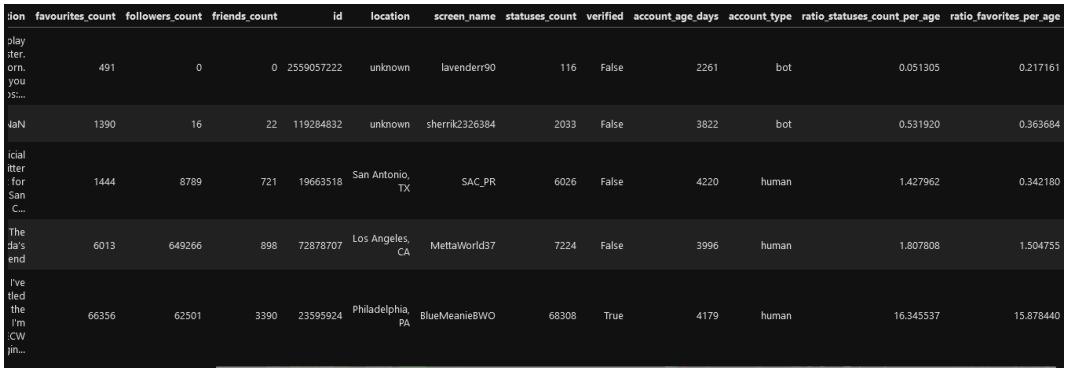
1. Parameter *ratio_statuses_count_per_age* digunakan untuk memberikan informasi tentang aktivitas dan intensitas pengguna di *Twitter* sehubungan dengan usianya. Parameter *ratio_statuses_count_per_age* mengacu pada *rasio* antara jumlah *tweet* yang diposting oleh pengguna dan usia pengguna tersebut. Dengan membandingkan jumlah *tweet* yang diposting dengan usia pengguna, dapat melihat sejauh mana pengguna aktif dalam berinteraksi dan berkontribusi di platform *Twitter*. Semakin besar nilai *ratio* maka dapat disimpulkan bahwa *profile* pengguna dikatakan aktif berinteraksi dengan platform *Twitter*, jika nilai kecil bahkan dibawah 0 maka *profile* pengguna dikatakan jarang aktif. Ini yang menjadi ciri dari akun bot dimana interaksi terhadap platform tidak tinggi. Perhitungan pada *source code* dapat dilihat pada Lampiran 8. Dan hasil perhitungan parameter akan langsung ditambahkan ke dataset yang dapat dilihat pada Gambar 4.6.



default_profile	description	favourites_count	followers_count	friends_count	id	location	screen_name	statuses_count	verified	account_age_days	account_type	ratio_statuses_count_per_age
True	Cosplay master. Unicorn. Waiting you at https:...	491	0	0	2559057222	unknown	lavenderr90	116	False	2261	bot	0.051305
True	NaN	1390	16	22	119284832	unknown	sherrik2326384	2033	False	3822	bot	0.531920
True	The official Twitter account for San Antonio C...	1444	8789	721	19663518	San Antonio, TX	SAC_PR	6026	False	4220	human	1.427962
False	The Panda's Friend	6013	649266	898	72878707	Los Angeles, CA	MettaWorld37	7224	False	3996	human	1.807808
False	I've wrestled for the #WWE. I'm an #ECW Origin...	66356	62501	3390	23595924	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human	16.345537

Gambar 4.6 Parameter *ratio_statusess_count_per_age*

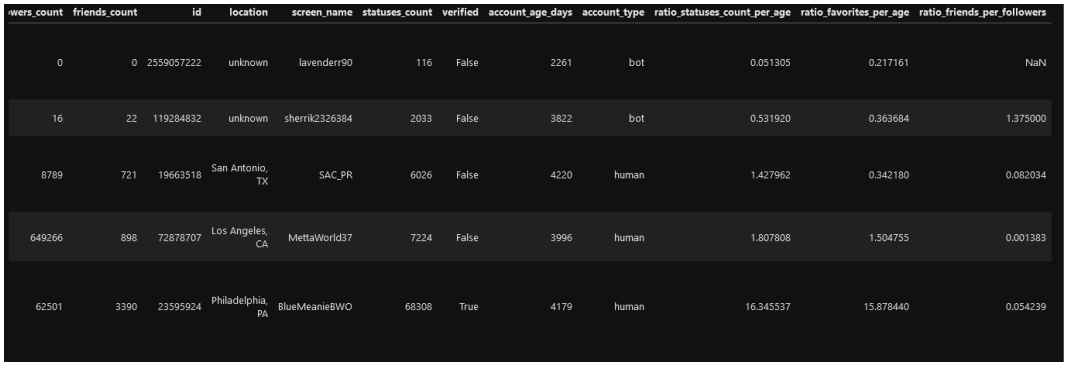
2. *Ratio_favourites_per_age* digunakan untuk menunjukkan *rasio* antara jumlah *tweet* yang ditandai sebagai jumlah favorit oleh pengguna dengan usia pengguna. ini memberikan informasi tentang sejauh mana pengguna tertarik dengan konten yang mereka temui di platform *Twitter* berdasarkan kelompok usia mereka. Perhitungan *source code* dapat dilihat pada Lampiran 9. Semakin besar nilai ratio favorit pengguna maka menunjukkan semakin populer juga pengguna menerima banyak favorit dari pengikutnya, dapat dilihat pada Gambar 4.7 dimana *ratio* favorit *human* cenderung lebih banyak yang menunjukkan data teratas yang terlabeli sebagai human benar benar akun pengguna aktif karena lebih tinggi nilainya dibandingkan dengan akun yang terlabeli sebagai bot.



ion	favourites_count	followers_count	friends_count	id	location	screen_name	statuses_count	verified	account_age_days	account_type	ratio_statuses_count_per_age	ratio_favorites_per_age
play iter. orn. you s:...	491	0	0	2559057222	unknown	lavenderr90	116	False	2261	bot	0.051305	0.217161
laN	1390	16	22	119284832	unknown	sherrik2326384	2033	False	3822	bot	0.531920	0.363684
icial itter for San C...	1444	8789	721	19663518	San Antonio, TX	SAC_PR	6026	False	4220	human	1.427962	0.342180
The da's end	6013	649266	898	72878707	Los Angeles, CA	MettaWorld37	7224	False	3996	human	1.807808	1.504755
I've tled the I'm CW jin...	66356	62501	3390	23595924	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human	16.345537	15.878440

Gambar 4.7 Parameter *ratio_favourites_per_age*

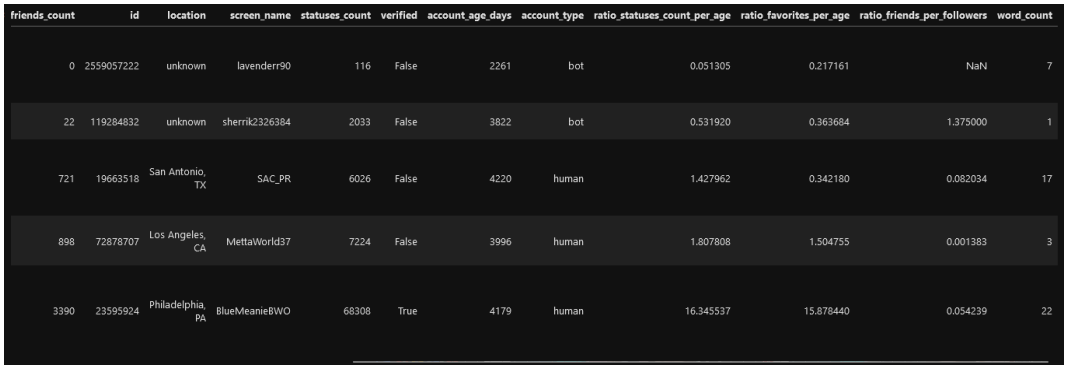
3. *Ratio_friends_per_followers* digunakan untuk menunjukkan *rasio* antara jumlah teman dengan jumlah pengikut pada *Twitter*. Perhitungan *source code* parameter R*atio_friends_per_followers* dapat dilihat pada Lampiran 10. Pada Gambar 4.8 dimana semakin besar rationya berarti jumlah teman yang dimiliki oleh pengguina lebih tinggi dibandingkan penginkutnya itu menunjukkan bahwa akun pengguna tersebut adalah akun yang lebih aktif berinteraksi seperti pada label bot lebih tinggi nilainya yang menunjukkan pengguna bot cenderung aktif berinteraksi dengan pengguna lain dan mengikuti pengguna lain.



wers_count	friends_count	id	location	screen_name	statuses_count	verified	account_age_days	account_type	ratio_statuses_count_per_age	ratio_favorites_per_age	ratio_friends_per_followers
0	0	2559057222	unknown	lavenderr90	116	False	2261	bot	0.051305	0.217161	NaN
16	22	119284832	unknown	sherrik2326384	2033	False	3822	bot	0.531920	0.363684	1.375000
8789	721	19663518	San Antonio, TX	SAC_PR	6026	False	4220	human	1.427962	0.342180	0.082034
649266	898	72878707	Los Angeles, CA	MettaWorld37	7224	False	3996	human	1.807808	1.504755	0.001383
62501	3390	23595924	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human	16.345537	15.878440	0.054239

Gambar 4.8 Parameter *ratio_friends_per_follower*

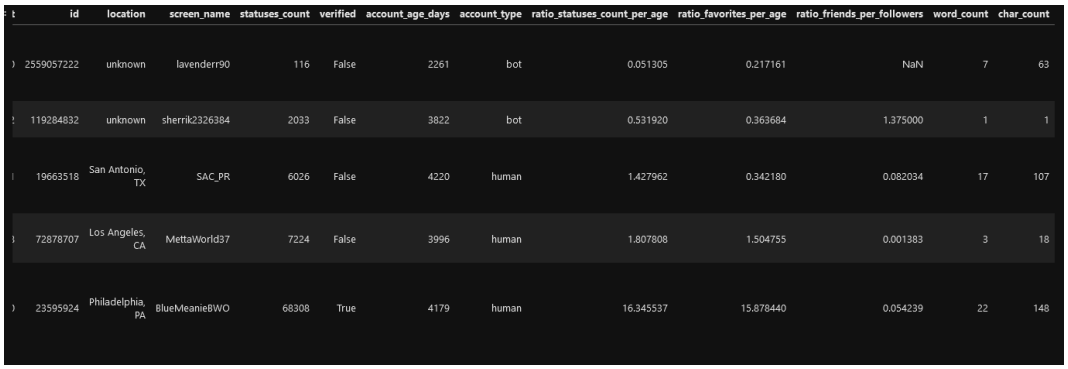
4. Parameter *word_count* digunakan untuk menghitung jumlah kata dalam sebuah *tweet*. ini membantu mengidentifikasi seberapa panjang atau pendek *tweet* tersebut berdasarkan jumlah kata yang digunakan. Perhitungan *source code* parameter *word_count* dapat dilihat pada Lampiran 11. Dalam kasus berikut contoh 1 data dengan paramter deskripsi “Cosplay master. Waiting you at https://t.co/YlP1D4EbFc” dimana terdapat 7 kata dari parameter tersebut. Parameter *word_count* yang sudah ditambahkan dapat dilihat pada Gambar 4.9.



friends_count	id	location	screen_name	statuses_count	verified	account_age_days	account_type	ratio_statuses_count_per_age	ratio_favorites_per_age	ratio_friends_per_followers	word_count
0	2559057222	unknown	lavenderr90	116	False	2261	bot	0.051305	0.217161	NaN	7
22	119284832	unknown	sherrik2326384	2033	False	3822	bot	0.531920	0.363684	1.375000	1
721	19663518	San Antonio, TX	SAC_PR	6026	False	4220	human	1.427962	0.342180	0.082034	17
898	72878707	Los Angeles, CA	MettaWorld37	7224	False	3996	human	1.807808	1.504755	0.001383	3
3390	23595924	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human	16.345537	15.878440	0.054239	22

Gambar 4.9 Parameter *Word Count*

5. Parameter *char_count* digunakan untuk menghitung jumlah karakter dalam sebuah *twit.* Perhitungan *source code* parameter *char_count* dapat dilihat pada Lampiran 12. Dimana pada Gambar 4.10 dengan data yang sama pada Gambar 4.8 parameter char_count ditambahkan berdasarkan jumlah *character* dari total parameter *description*.



id	location	screen_name	statuses_count	verified	account_age_days	account_type	ratio_statuses_count_per_age	ratio_favorites_per_age	ratio_friends_per_followers	word_count	char_count
2559057222	unknown	lavenderr90	116	False	2261	bot	0.051305	0.217161	NaN	7	63
119284832	unknown	sherrik2326384	2033	False	3822	bot	0.531920	0.363684	1.375000	1	1
19663518	San Antonio, TX	SAC_PR	6026	False	4220	human	1.427962	0.342180	0.082034	17	107
72878707	Los Angeles, CA	MettaWorld37	7224	False	3996	human	1.807808	1.504755	0.001383	3	18
23595924	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human	16.345537	15.878440	0.054239	22	148

Gambar 4.10 Parameter *char_count*

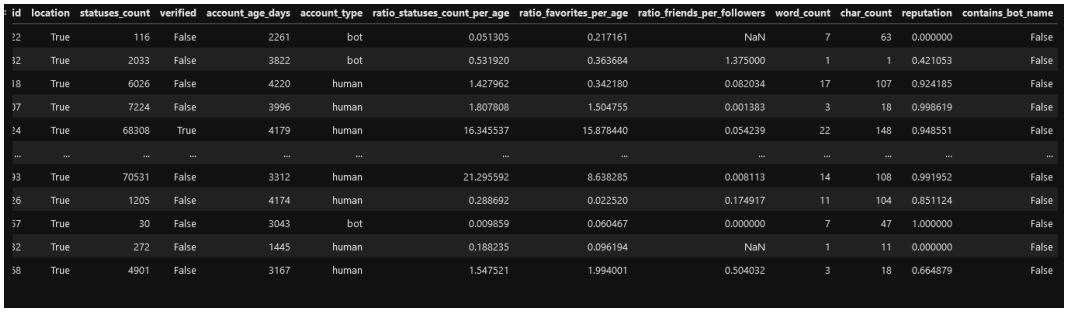
6. Parameter *Reputation* digunakan untuk ukuran tingkat reputasi pengguna *Twitter*. ini mencerminkan pengguna dilihat oleh orang lain berdasarkan aktivitas dan kontribusinya. Perhitungan *source code* parameter reputation dapat dilihat pada Lampiran 13. Hasil parameter *reputation* ditambahkan pada dataset dan dapat dilihat pada Gambar 4.11.

d	location	screen_name	statuses_count	verified	account_age_days	account_type	ratio_statuses_count_per_age	ratio_favorites_per_age	ratio_friends_per_followers	word_count	char_count	reputation
2	unknown	lavenderr90	116	False	2261	bot	0.051305	0.217161	NaN	7	63	NaN
2	unknown	sherrik2326384	2033	False	3822	bot	0.531920	0.363684	1.375000	1	1	0.421053
8	San Antonio, TX	SAC_PR	6026	False	4220	human	1.427962	0.342180	0.082034	17	107	0.924185
7	Los Angeles, CA	MettaWorld37	7224	False	3996	human	1.807808	1.504755	0.001383	3	18	0.998619
4	Philadelphia, PA	BlueMeanieBWO	68308	True	4179	human	16.345537	15.878440	0.054239	22	148	0.948551

Gambar 4.11 Parameter *reputation*

Pada Gambar 4.11, dapat dilihat bahwa reputasi dengan label *human* cenderung lebih baik dibandingkan dengan akun yang terlabeli bot. Reputasi yang mendekati nilai 1 adalah akun yang berpengaruh. Di sisi lain, pengguna dengan jumlah pengikut yang sedikit dan banyak mengikuti pengguna lain akan memiliki reputasi yang mendekati nilai 0 seperti data pada label bot diatas.

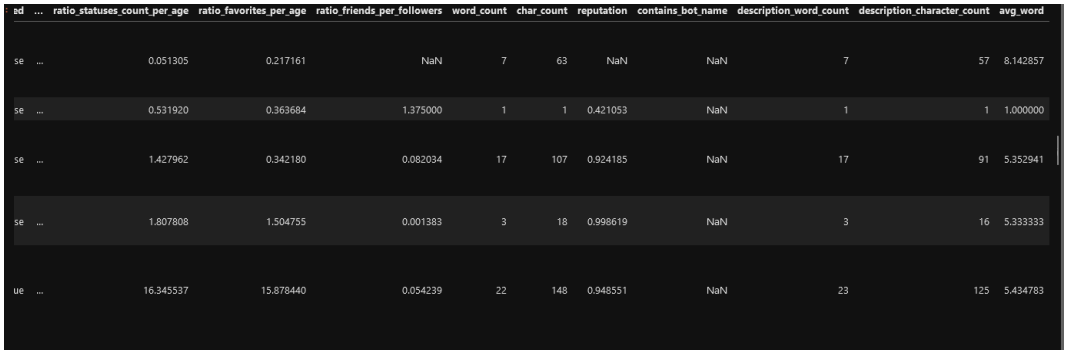
7. Parameter *contains_bot_name* digunakan untuk menentukan apakah suatu deskripsi mengandung kata-kata yang menunjukkan adanya bot. Jika deskripsi pada *field* data tersebut mengandung kata "bot", maka parameter akan mengembalikan nilai *True*. Sebaliknya, jika deskripsi tidak mengandung kata-kata tersebut, parameter akan mengembalikan nilai *False*. Karena beberapa akun bot sering mencantumkan teks “bot” pada bio mereka untuk memperkenalkan langsung kepada pengguna lain bahwa contoh akun ini adalah bot. *source code* untuk menghitung dan menambahkan parameter *contains_bot_name* dapat dilihat pada Lampiran 7. Data yang sudah hitung akan langsung di tambahkan dan dapat dilihat pada Gambar 4.12.



id	location	statuses_count	verified	account_age_days	account_type	ratio_statuses_count_per_age	ratio_favorites_per_age	ratio_friends_per_followers	word_count	char_count	reputation	contains_bot_name
22	True	116	False	2261	bot	0.051305	0.217161	NaN	7	63	0.000000	False
32	True	2033	False	3822	bot	0.531920	0.363684	1.375000	1	1	0.421053	False
18	True	6026	False	4220	human	1.427962	0.342180	0.082034	17	107	0.924185	False
07	True	7224	False	3996	human	1.807808	1.504755	0.001383	3	18	0.998619	False
24	True	68308	True	4179	human	16.345537	15.878440	0.054239	22	148	0.948551	False
...	...	...	...	...	...	...	...	...	...	...	...	...
93	True	70531	False	3312	human	21.295592	8.638285	0.008113	14	108	0.991952	False
26	True	1205	False	4174	human	0.288692	0.022520	0.174917	11	104	0.851124	False
57	True	30	False	3043	bot	0.009859	0.060467	0.000000	7	47	1.000000	False
32	True	272	False	1445	human	0.188235	0.096194	NaN	1	11	0.000000	False
58	True	4901	False	3167	human	1.547521	1.994001	0.504032	3	18	0.664879	False

Gambar 4.12 Parameter *contains_bot_name*

8. Parameter *avg_word* digunakan untuk menghitung rata-rata panjang kata dalam sebuah *tweet*. *source code* dapat dilihat pada Lampiran 15.



ed	...	ratio_statuses_count_per_age	ratio_favorites_per_age	ratio_friends_per_followers	word_count	char_count	reputation	contains_bot_name	description_word_count	description_character_count	avg_word
se	...	0.051305	0.217161	NaN	7	63	NaN	NaN	7	57	8.142857
se	...	0.531920	0.363684	1.375000	1	1	0.421053	NaN	1	1	1.000000
se	...	1.427962	0.342180	0.082034	17	107	0.924185	NaN	17	91	5.352941
se	...	1.807808	1.504755	0.001383	3	18	0.998619	NaN	3	16	5.333333
ue	...	16.345537	15.878440	0.054239	22	148	0.948551	NaN	23	125	5.434783

Gambar 4.13 Parameter *avg_word*

Dalam hasil data yang ada pada Gambar 4.13, jika data yang muncul semakin besar maka menunjukkan bahwa rata – rata Panjang yang diambil dalam sebuah parameter *description* lebih besar dibandingkan parameter *description* akun yang lain. Pada data yang terlabeli human, nilai yang didapat lebih besar daripada data yang terlabeli bot. ini menunjukkan jika pengguna aktif *human* menggunakan kata kata yang lebih Panjang.

4.3.3 *Desk Checking*

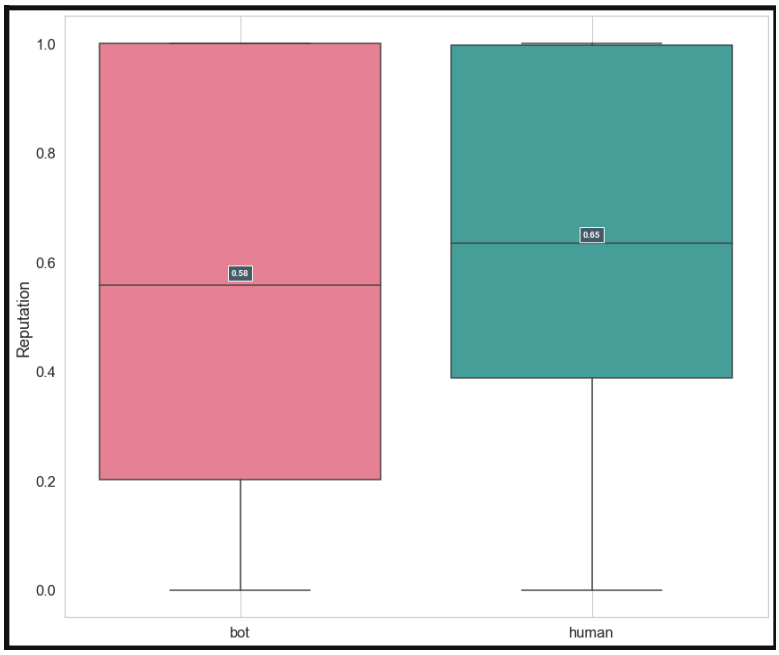
Desk Checking digunakan untuk proses meninjau kode sumber program secara manual dan salah satu tahapan yang mempunyai kepentingan tinggi. *Desk checking* berfungsi untuk mengidentifikasi kesalahan logika, kesalahan sintaksis, atau kesalahan pemrograman lainnya sebelum kode dijalankan secara aktif. Dalam

desk checking, pengujian biasanya dilakukan dengan melihat kode secara perlahan dan memastikan bahwa setiap langkah atau instruksi dijalankan dengan benar dan menghasilkan *output* yang diharapkan Vendor perangkat lunak yang telah mempelajari bahasa pemrograman dengan sangat baik akan sering diikutsertakan dalam tes *Desk Checking*. (Rizaldi et al., 2022)

Pada penelitian ini dilakukan checking terhadap kode yang sudah di gunakan dan dibandingkan hasil yang didapat secara hitungan manual yang dapat dilihat pada Tabel 4.3. Hasil perhitungan yang dilakukan oleh sistem terhadap parameter – parameter yang digunakan bernilai benar dan dapat dilakukan analisa selanjutnya.

4.3.4 *Explanatory Data Analysis (EDA)*

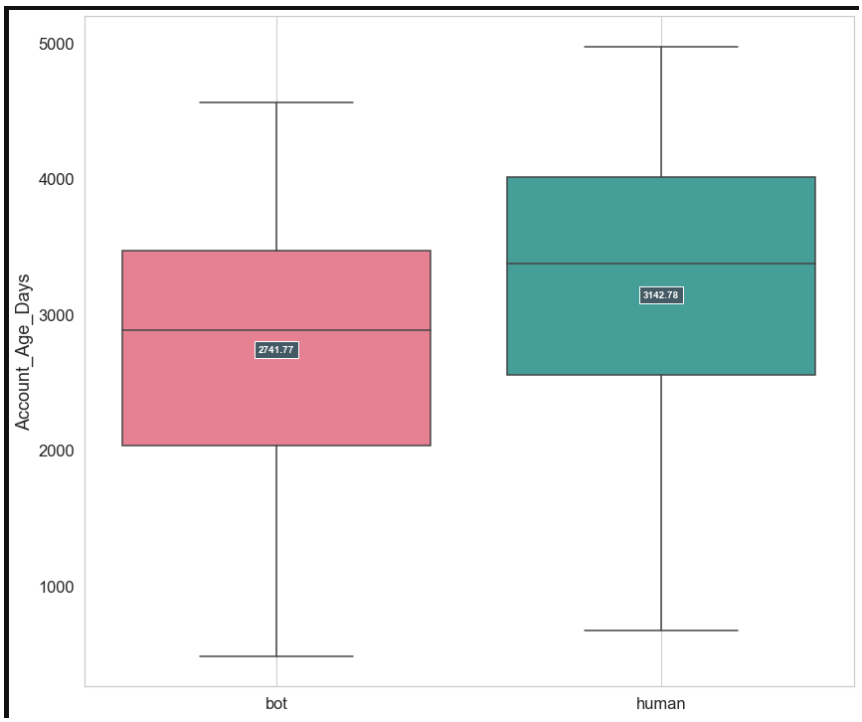
Setelah melakukan proses *preprocessing data*, selanjutnya data dapat divisualkan lebih jelas dengan menggambarkan beberapa parameter. Visualisasi data menggunakan *boxplot* dengan *multiple boxplot* dari beberapa parameter yang menjadi nilai acuan.



Gambar 4.14 *boxplot* parameter *Reputation*

Boxplot berikut dapat dilihat pada Gambar 4.14 dimana data dari parameter *reputation*, tujuannya adalah melihat seberapa banyak perbedaan reputasi dari

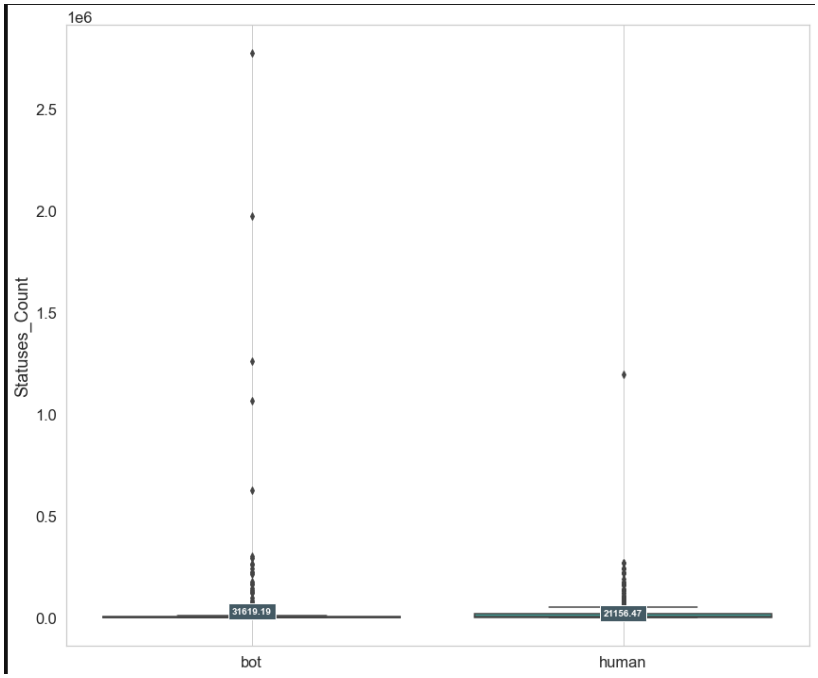
semua akun yang berlabelkan bot dan human. parameter grafik berwarna merah sebagai bot dan hijau sebagai *human*. Data tersebut menunjukkan bahwa pada diagram bot dan *human* sama sama memiliki reputasi yang sangat kuat dimana diagram keduanya hampir mendekati nilai 1. Namun, pada diagram akun yang berlabel bot terdapat reputasi yang tidak seimbang dimana sumbu diagram bot lebih dominan kecil dari pada sumbu diagram *human*, sedangkan hasil reputasi akun *human* menunjukkan cenderung lebih baik dan seimbang dibandingkan dengan akun yang berlabelkan bot.



Gambar 4.15 *boxplot age_in_days*

Boxplot selanjutnya diambil dari *parameter age_in_days* yaitu usia akun dalam hari. Dapat dilihat pada Gambar 4.15 bahwa terdapat perbedaan data dari kedua kelas, terlihat dimana pada diagram berwarna merah yang berlabelkan bot mempunyai nilai 2741 yang artinya adalah kebanyakan akun yang dibuat sekitar 2000 hingga 2700 hari yang lalu, sedangkan pada diagram berwarna hijau yang berlabelkan *human* terlihat nilai 3233 yang artinya kebanyakan akun yang di dibuat 2000 hingga 3000 hari yang lalu atau sekitar 5 hingga 8 tahun yang lalu.

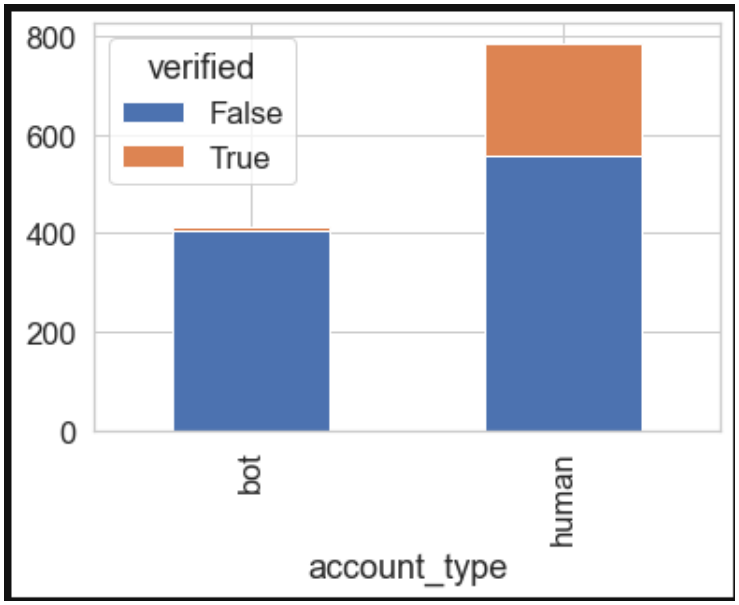
Dari data tersebut dapat disimpulkan bahwa data ini menunjukkan kemunculan akun akun bot akhir-akhir ini baru muncul.



Gambar 4.16 *bloxplot statuses count*

Boxplot selanjutnya diambil dari parameter *statuses_count* yang terlihat pada Gambar 4.16 dimana visualisasi *statuses_count* menunjukkan bahwa diagram data yang berlabel bot lebih tinggi dengan jumlah status yang serupa dibandingkan diagram data yang berlabelkan *human*. Dalam hal ini, peningkatan frekuensi pada diagram bot mengidentifikasikan adanya aktivitas yang tidak manusiawi di platform tersebut. Karena dengan jumlah status yang serupa dan dengan frekuensi tinggi sebuah akun dapat dikatakan sebagai *spammer*.

Akun *spammer* cenderung melakukan aktivitas yang berlebihan, seperti mengirim banyak pesan otomatis. (Aditya et al., 2019)



Gambar 4.17 *account type count*

Lalu pada visual selanjutnya adalah *boxplot* dari parameter *verified* yang ada pada Gambar 4.17 dimana dapat disimpulkan bahwa diagram dengan warna biru pada label *human* yaitu akun yang di lebelkan sebagai *human* namun belum *verified* dan warna *orange* adalah akun yang terlabelkan sebagai *human* sudah *verified*. Begitupun pada diagram dengan kelas bot dimana pada warna biru terdapat akun yang tidak *verified* terlabeli sebagai bot dan sedikit akun yang terlabeli sebagai bot namun sudah verified oleh *Twitter*. Ini menunjukkan bahwa pola akun yang diidentifikasi sebagai *human* kebanyakan *verified* karena verifikasi akun pada *Twitter* sebagai tanda pengenal resmi untuk mengkonfirmasi keaslian akun. Sedangkan banyak akun bot yang hanya dibuat tanpa mementingkan tanda pengenal.

4.3.5 Statistika Deskriptif

Statistika deskriptif digunakan dalam merangkum dan meringkas karakteristik kunci dari dataset, seperti rata-rata, median, modus, kuartil, dan rentang. Setelah proses *preprocessing* selesai, *dataset* yang telah dipersiapkan dapat melanjutkan dengan implementasi algoritma, namun perlu mengidentifikasi terlebih dahului karakteristik data menggunakan metode statistika deskriptif.

Untuk melakukan hal ini, dapat menggunakan *source code* yang dapat dilihat pada Lampiran 16.

Metode statistik deskriptif digunakan untuk menghitung statistik seperti *count* (jumlah data), *mean* (rata-rata), *std* (standar deviasi), *min* (nilai minimum), 25% (kuartil pertama), 50% (median), 75% (kuartil ketiga), dan *max* (nilai maksimum) dari setiap kolom dalam *dataset*. Namun, ada beberapa parameter yang dikecualikan karena tidak dapat dianalisis menggunakan metode statistika deskriptif karena merupakan data *string* dan *Boolean*. Dalam visualisasi datanya dapat dilihat pada Tabel 4.4.

Tabel 4.4 Statistika Deskriptif setelah data di *preprocessing*

Nama Parameter	Mean	Standar Deviasi	Min	Median	Max
favourites_count	9885,949167	24747,54362	0	2057,5	373633
followers_count	464021,5875	3512307,087	0	300	98161375
friends_count	4918,279167	43323,73082	0	263	933476
statuses_count	21090,94	63064,60654	0	3945	1258100
account_age_days	3004,428333	1020,857515	483	3190,5	4972
ratio_statuses_count_per_age	6,638758528	18,6933062	0	1,418647408	337,2018226
ratio_favorites_per_age	3,764408148	10,86666805	0	0,64928771	185,8958525
ratio_friends_per_followers	2,303236456	8,128659033	0	0,641612201	151,7142857
word_count	10,00833333	8,263282162	1	8	68

Lanjutan Tabel 4.4 Statistika Desktiptif setelah data di *preprocessing*

char_count	66,40916667	54,1151284	1	57	162
Reputation	0,625996425	0,347881981	0	0,609157483	1
description_word_count	9,968333333	8,100329829	1	8	34
description_character_count	57,26	46,50514	1	49	147
avg_word	5,516999	4,875347	1	5,25	63

Data sebelum dilakukan *scale* atau normalisasi menghasilkan nilai *min* dan *max* yang tidak stabil. Ini membuat data skala tidak seimbang dan membuat salah satu parameter akan memiliki pengaruh yang dominan saat melakukan analisa menggunakan algoritma. Untuk mengatasi rentang yang besar antara nilai maksimum dan minimum dalam *dataset*, digunakan metode skalabilitas *MinMaxScaler()* dari *libray sklearn*. Metode *MinMaxScaler* berfungsi untuk mengubah nilai setiap fitur secara individu sehingga rentang nilainya berada dalam kisaran yang telah ditentukan. (Ambarwari et al., 2020)

Metode ini tidak mengubah distribusi data fitur menjadi distribusi normal seperti halnya pada normalisasi. Scaling hanya mempertahankan distribusi data yang ada. contoh kode sumber yang digunakan untuk menerapkan metode skalabilitas ini dapat dilihat pada Lampiran 17. Dengan menggunakan *MinMaxScaler* data pada dataset dengan rentang nilai data menjadi rentang yang ditentukan, dalam hal ini *dataset* di *scale* menjadi dari 0 hingga 100, Hal ini memastikan bahwa data yang di dapat seimbang dan dapat diolah baik oleh algoritma *Random Forest*. Selanjutnya, Hasil transformasi disimpan dan menghasilkan data baru dengan format *.xlsx* yang merupakan data yang telah diubah skala sesuai dengan metode *Min-Max* Scaling dengan nilai yang sudah ditentukan menggunakan objek *scale*. Hasil data dapat dilihat pada Lampiran 18

4.4 Implementasi Random Forest

Implementasi algoritma *Random Forest Classifier* membutuhkan *library* dari *scikit-learn*, *pandas*, dan *numpy*. Dari *dataset* yang sudah di normalisasi menggunakan *scale* dapat dipilih *feature* atau parameter dan label yang akan digunakan pada *Random Forest*. *Dataset* lalu dibagi menjadi *training dataset* dan *testing dataset*. Untuk melakukan pembagian ini, dapat dilihat pada Lampiran 19.

```
Akurasi model Random Forest: 0.8541666666666666
```

Gambar 4.18 Akurasi *Random Forest*

Proses pembuatan model *Random Forest* dibuat dengan menggunakan 100 pohon keputusan dan Selama pelatihan, model akan mempelajari pola dan hubungan antara *feature* dan label pada data latih sebanyak 960 untuk mendapatkan prediksi yang akurat. Setelah proses analisa didapat akurasi yang dapat dilihat pada Gambar 4.18.

Tabel 4.5 perbedaan akurasi pada ratio partisi

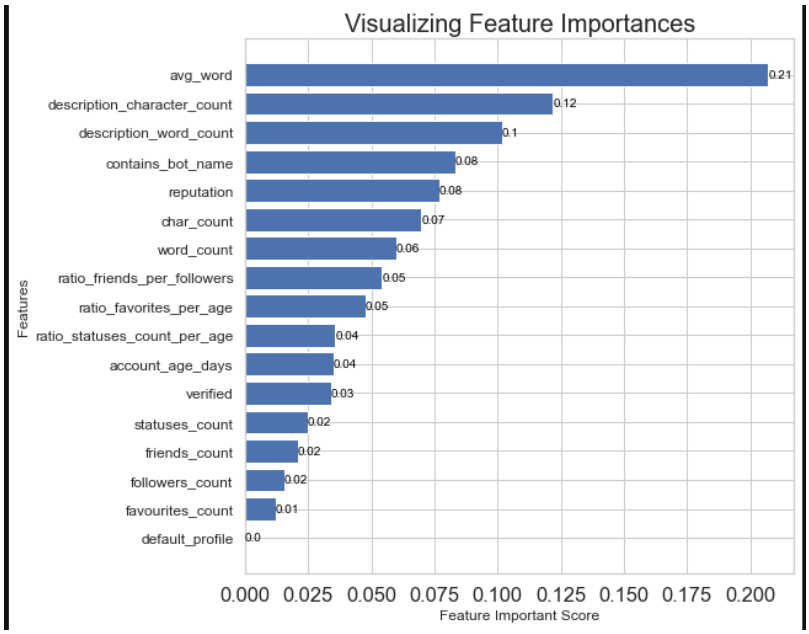
Ratio Partisi	Hasil Akurasi
50:50	81.84%
60:40	82.71%
70:30	83.34%
80:20	85.42%
90:10	81.67%

Berdasarkan hasil pengujian, data yang digunakan algoritma *Random Forest* di uji dengan *ratio* yang berbeda. Dari beberapa *ratio* yang digunakan *ratio partision* yang menghasilkan akurasi paling tinggi didapatkan dengan *ratio* 80:20 dengan akurasi sebesar 85%. Hasil data dapat dilihat pada Tabel 4.5.

Selanjutnya proses penyimpanan model pembelajaran dari *Random Forest* menggunakan *library joblib* dari *scikit-learn* untuk menghasilkan model menjadi file *model.pkl*. Model kemudian digunakan untuk memprediksi data pada *dataset* yang ada pada sistem deteksi tanpa perlu melatih *dataset* lagi.

4.4.1 Skor *Feature Importance*

Setelah proses implementasi algoritma, selanjutnya adalah melihat pengaruh setiap *feature* terhadap akurasi model klasifikasi deteksi bot menggunakan skor kepentingan *feature*. Skor kepentingan *feature* digunakan sebagai panduan dalam melakukan seleksi *feature* guna mencapai akurasi tertinggi berdasarkan parameter yang ada.



Gambar 4.19 Diagram *Feature Importance*

Berdasarkan grafik pada Gambar 4.19 diidentifikasi bahwa hasil parameter *avg_word* memiliki skor kepentingan paling tinggi dengan nilai 0,21. Setelah itu baru dilanjutkan dengan parameter *description_character_count*, dan *description_word_count*. Hal ini menunjukkan bahwa parameter tersebut sangat memiliki bobot paling tinggi dalam pertimbangan klasifikasi pada *dataset*. Sebaliknya parameter *default_profile* memiliki skor kepentingan paling rendah dengan nilai 0.00. Hal ini menunjukkan bahwa parameter tersebut memiliki bobot paling sedikit dalam pertimbangan menentukan klasifikasi pada data. Nilai data Dari skor *feature* dapat dilihat secara rinci pada Tabel 4.6.

Tabel 4.6 Skor *feature importance*

Nama *Feature*	Skor f1
Avg_word	0,21
Description_character_count	0,12
Description_word_count	0,1
Contains_bot_name	0,08
Reputation	0,08
Char_count	0,07
Word_count	0,06
Ratio_friends_per_followers	0,05
Ratio_favourites_per_age	0,05
Ratio_statuses_count_per_age	0,04
Accounts_age_days	0,04
Verified	0,03
Statuses_count	0,02
Friends_count	0,02
Followers_count	0,02
Favourites_count	0,01
Default_profile	0,00

4.5 Pengujian model Random Forest

Evaluasi model yang dihasilkan oleh algoritma *Random Forest Classifier* digunakan untuk tugas klasifikasi. Terdapat beberapa tahapan utama dalam proses evaluasi ini, termasuk pembuatan *confusion matrix*, pengukuran akurasi model secara keseluruhan, dan pembuatan laporan klasifikasi yang menyajikan akurasi, *presision*, *recall*, dan *f1-score*. Perhitungan *source code* dapat dilihat pada Lampiran 20. berdasarkan hasil pengujian menggunakan *confusion matrix* data yang dihasilkan berupa tabel yang dapat dilihat keseluruhan nilainya pada Tabel 4.7 dan Tabel 4.8.

Tabel 4.7 hasil nilai dari confusion matrix kelas bot dan human

actual	bot	65	23
	Human	12	140
		bot	human
		predicted	

Tabel 4.8 hasil dari evaluasi confusion matrix

actual	bot	TN	FP
	Human	FN	TP
		Bot	human
		Predicted	

Bedasarkan hasil dari visualisasi dari Tabel 4.8 data tersebut dapat didefinisikan *predicted True positive (TP)* adalah variabel yang menunjukkan prediksi *human* di klasifikasikan dengan benar sebagai human, *False positive (FP)* adalah variabel yang menunjukkan sampel *human* yang salah di klasifikasikan sebagai human, *True negative (TN)* adalah variabel yang menunjukkan sampel bot yang diklasifikasikan dengan benar sebagai bot, dan *False negative (FN)* adalah variabel yang menunjukkan sampel bot yang salah di klasifikasikan sebagai bot.

4.6 Evaluasi Model klasifikasi

proses terakhir yang dilakukan adalah menghasilkan evaluasi ringkasan model klasifikasi. Ringkasan klasifikasi ini didapat dengan menggunakan *source code* yang ada pada Lampiran 21. Hasil yang didapat adalah ringkasan evaluasi dari proses klasifikasi dapat dilihat pada Tabel 4.9.

Tabel 4.9 nilai laporan presisi,recall dan f1-score

		Precission	Recall	F1-score	Support
actual	Bot	0.84	0.74	0.79	88
	Human	0.86	0.92	0.89	152
			predicted		

Kelas yang memiliki presisi paling tinggi adalah human dengan nilai 0.86, sedangkan kelas bot hanya mempunyai nilai 0.84. Presisi adalah kemampuan *classifier* untuk tidak melabeli data *feature* positif ketika data *feature* tersebut sebenarnya negatif. Kelas yang memiliki *Recall* paling tinggi adalah human dengan nilai 0.92. Sedangkan bot hanya mempunyai nilai 0.74. Kelas yang memiliki *f1-score* paling tinggi adalah *human* dengan nilai 0.89. sedangkan bot hanya mempunyai nilai 0.79. dan *Support* merupakan jumlah sampel yang digunakan sebagai data uji. Jumlah *support* tiap-tiap kelas tidak sama satu dengan yang lainnya.

4.6.1 Proses *serialization*

Serialization bertujuan untuk menyimpan model sehingga dapat digunakan kembali untuk memprediksi data baru. Serialization ini dilakukan dengan menggunakan *library joblib* dan menyimpan file dalam format pickle (.pkl) hasil dapat dilihat pada Lampiran 22.

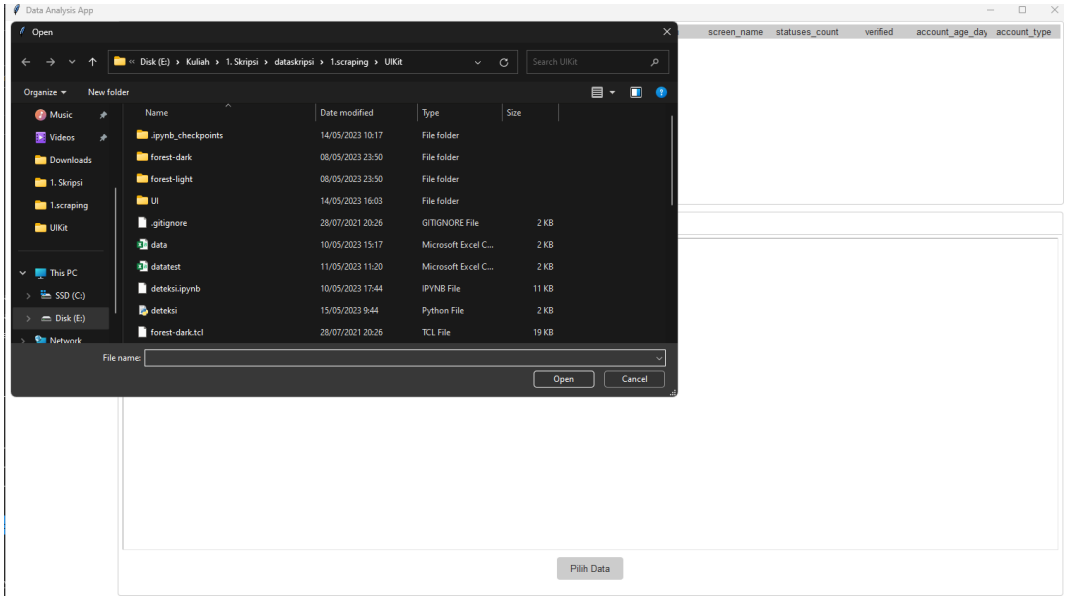
4.7 Implementasi Model ke dalam GUI

Implementasi *GUI (Graphical user interface)* digunakan untuk memvisualisasikan sekaligus menampilkan hasil prediksi terhadap datatest yang berupa *apps dekstop*. Dengan menggunakan *tkinter* sebagai modul untuk mempermudah pembuatan antarmuka. Terdapat beberapa halaman yang dapat dilihat sebagai berikut :

1. Halaman utama

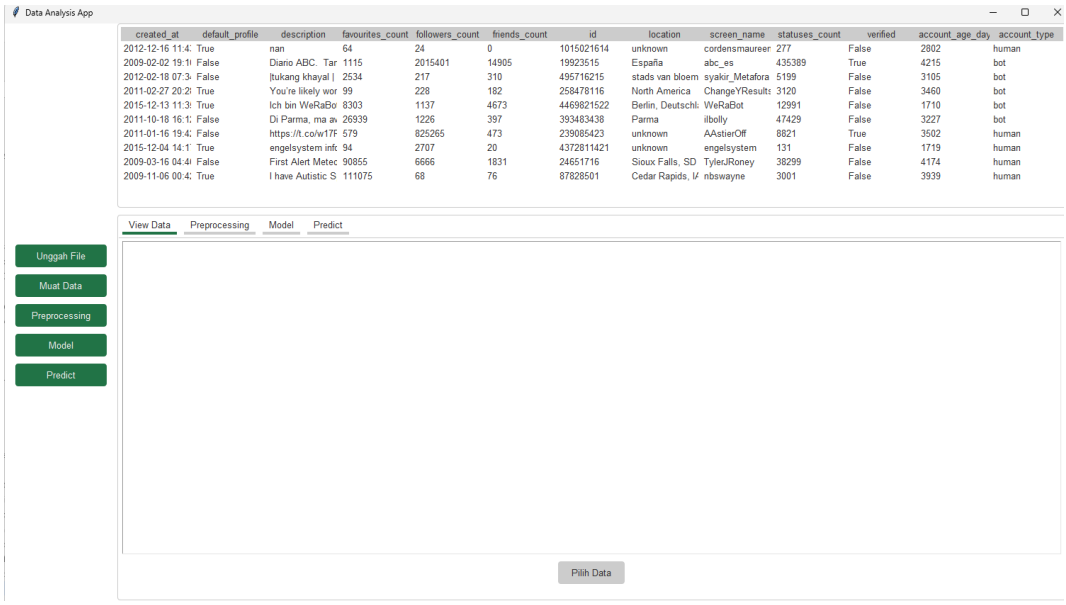
Sesuai dengan *wireframe* sebelumnya pada tampilan awal akan terdapat beberapa opsi *button* dan tab *view* masing – masing. Pada tahap awal pengguna

dapat mengklik *button* unggah file dan muat data untuk menampilkan hasil data yang telah diunggah. Hasil data yang tampil sebagai berikut :



Gambar 4.20 Unggah *dataset*

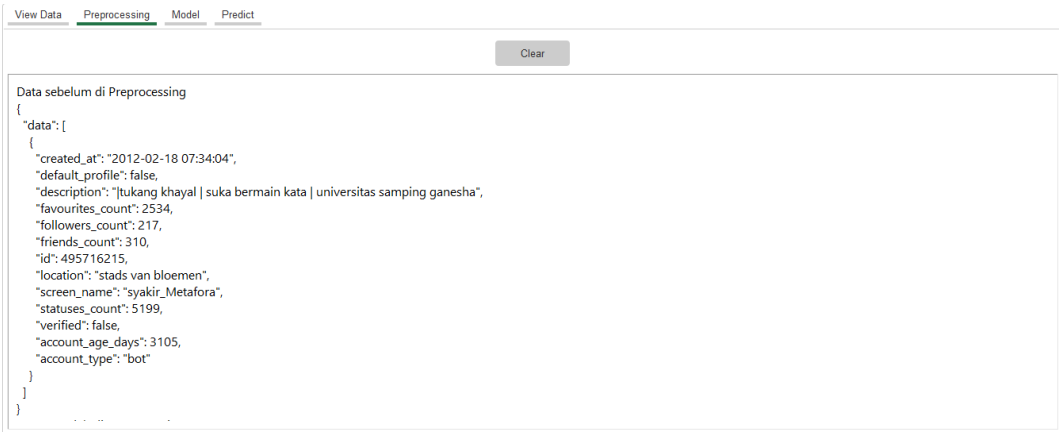
Data yang dipilih harus berformat .csv karena untuk mempermudah Sistem *GUI (Graphical User Interface)* dalam membaca dan menganalisa data nantinya.



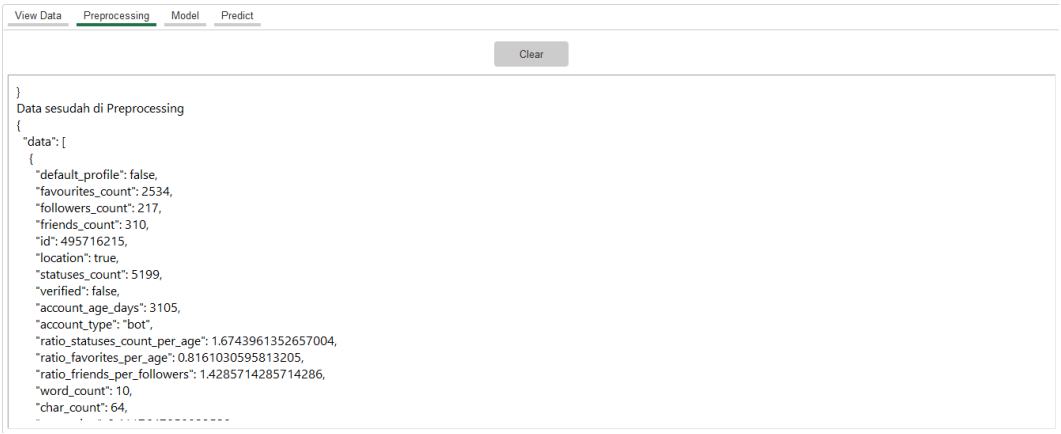
Gambar 4.21 Muat dataset

2. Halaman *preprocessing*

Setelah data dipilih secara random maka berikutnya melakukan *preprocessing*. pengguna dapat mengklik button *preprocessing* dibawah *button* muat data. Dan *output* yang dihasilkan berupa data sebelum dan sesudah di *preprocessing* yang dapat dilihat pada Gambar 4.22



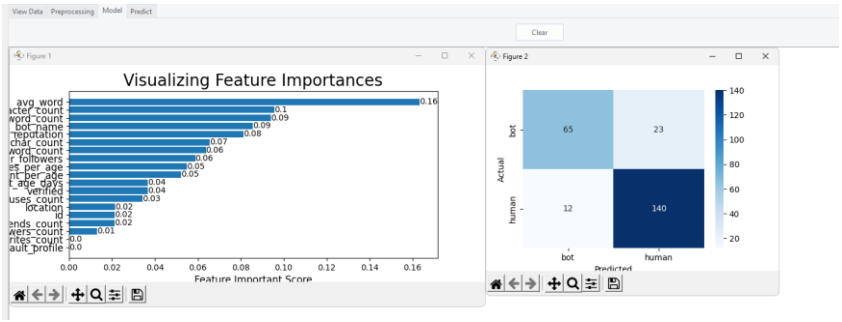
Gambar 4.22 Hasil data sebelum di *preprocessing*



Gambar 4.23 Hasil data sesudah di *preprocessing*

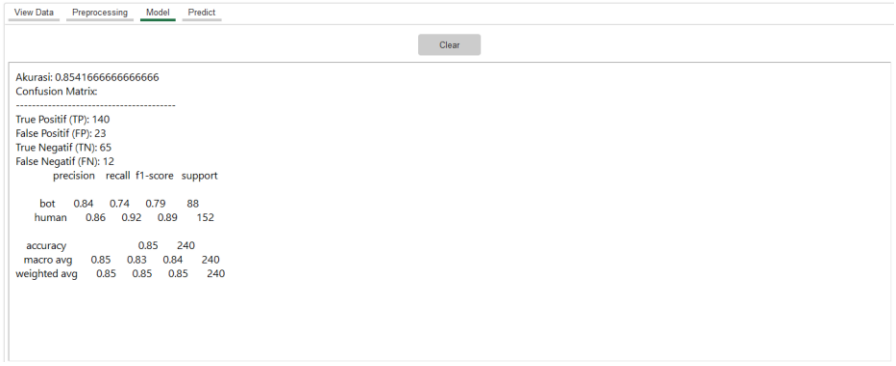
3. Halaman model

Pada halaman model algoritma *Random Forest* akan mulai berjalan. *Button* model terhubung dengan bagian pengodean atau logika di belakangnya memicu pemrosesan data dan pembentukan model *Random Forest*. Dengan menggunakan *button* model. *output* yang dihasilkan dapat dilihat pada Gambar 4.24.



Gambar 4.24 visualisai skor hasil model

Hasil visualisasi dapat dilihat pada Gambar 4.23 yang menunjukkan parameter yang digunakan untuk sistem deteksi memiliki bobot pengaruh yang berbeda beserta hasil pengujian model menggunakan confusion matrix yang menampilkan jumlah data uji yang di prediksi.



Gambar 4.25 Hasil dari Analisa model *Random Forest*

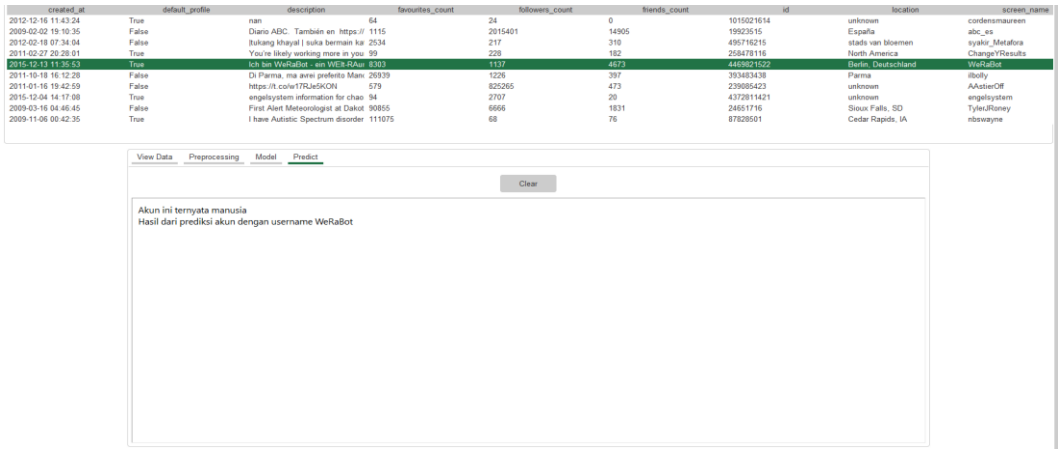
```
Jumlah Pohon Keputusan : 100
Nama Kelas: ['bot' 'human']
|--- ratio_favorites_per_age <= 1.33
|   |--- verified <= 0.50
|   |   |--- default_profile <= 0.50
|   |   |   |--- reputation <= 0.14
|   |   |   |   |--- account_age_days <= 1007.50
|   |   |   |   |   |--- weights: [0.00, 1.00] class: 1.0
|   |   |   |   |--- account_age_days >  1007.50
|   |   |   |   |   |--- ratio_friends_per_followers <= 8.00
|   |   |   |   |   |   |--- weights: [9.00, 0.00] class: 0.0
|   |   |   |   |   |--- ratio_friends_per_followers >  8.00
|   |   |   |   |   |   |--- reputation <= 0.08
|   |   |   |   |   |   |   |--- weights: [6.00, 0.00] class: 0.0
|   |   |   |   |   |   |--- reputation >  0.08
|   |   |   |   |   |   |   |--- weights: [0.00, 1.00] class: 1.0
```

Gambar 4.26 Hasil pohon keputusan *Random Forest*

Hasil dari 100 pohon keputusan model *Random Forest* dapat dilihat pada Gambar 4.25. Nilai parameter pembanding didapat dari pelatihan model, Dimana semua parameter – parameter yang digunakan seperti '*ratio_favorites_per_age*' kondisinya benar maka akan melanjutkannya ke *node* berikutnya. Hasil akhir dari pohon keputusan adalah *wight* yang berarti hasil voting dapat masuk klasifikasi *class* 1 berarti *human* atau *class* 2 yang berarti bot.

4. Halaman prediksi

Halaman ini merupakan halaman yang menampilkan hasil klasifikasi dari data yang dipilih. *Output* yang keluar berupa *username* dan label. Berikut tampilan hasil deteksi untuk *username @WeRaBot* dengan hasil klasifikasi manusia (*human*). Hasil tersebut adalah proses prediksi dari data yang pengguna milih dan model *random forest* yang sudah di dapat. Kemudian data tersebut dilakukan *voting* apakah data yang sudah dipilih sesuai dengan hasil Analisa model. Maka *output* yang dihasilkan dapat berupa "akun ini bot tetapi pada palebelan sebelumnya dilabelkan sebagai manusia" atau "akun ini manusia tetapi pada palebelan sebelumnya dilabelkan sebagai bot".



Gambar 4.27 Hasil prediksi aplikasi deteksi bot

Dari hasil sistem deteksi menggunakan *GUI (Graphical User Interface)* tersebut dapat digunakan berulang kali untuk mendeteksi sebuah pengguna dengan menggunakan data *username Twitter* yang berbeda. Sistem juga dapat digunakan untuk melatih *dataset* yang lebih besar agar hasil akurasi yang di dapat meningkat

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian terhadap klasifikasi bot *Twitter* dengan mengguinakan *Random Forest Classifier* dapat disimpulkan bahwa algoritma *Random Forest* dapat digunakan sebagai metode untuk mendeteksi bot pada platform *Twitter* dengan menggunakan sistem *GUI (Graphical User Interface)* yang menghasilkan label pada 10 data akun pengguna yang diuji. Dengan menggunakan metode ini, sistem deteksi bot mampu membedakan antara akun bot dan akun non-bot dengan tingkat akurasi yang baik dan mendapatkan akurasi paling tinggi (85.42%).

5.2 Saran

Berdasarkan penelitian ini, perlu ditambahkan beberapa saran yang harus dipertimbangkan dalam pengembangan penelitian selanjutnya :

1. Jika penelitian ini akan dikembangkan maka pada perlu implementasi pada website agar mempermudah dalam mengaksesnya.
2. Perlu metode oversampling atau undersampling untuk menyeimbangkan data imbalance agar akurasi yang didapat lebih besar.

DAFTAR PUSTAKA

Adhisyanda, A. (2020). *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS) Penggabungan Teknologi Untuk Analisa Data Berbasis Data Science*. 50–51.

Adi Yahyadi. (2022). ANALISIS SENTIMEN TWITTER TERHADAP KEBIJAKAN PPKM DI TENGAH PANDEMI COVID-19. *Journal of Information System, Applied, Management, Accounting and Research.*, *6*(2), 464–471. https://doi.org/10.52362/jisamar.v6i2.791

Aditya, C. S. K., Hani'ah, M., Fitrawan, A. A., Arifin, A. Z., & Purwitasari, D. (2019). Deteksi Bot Spammer pada Twitter Berbasis Sentiment Analysis dan Time Interval Entropy. *Jurnal Buana Informatika*, *7*(3), 179–186. https://doi.org/10.24002/jbi.v7i3.656

Ambarwari, A., Jafar Adrian, Q., & Herdiyeni, Y. (2020). Analysis of the Effect of Data Scaling on the Performance of the Machine Learning Algorithm for Plant Identification. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *4*(1), 117–122. https://doi.org/10.29207/resti.v4i1.1517

Azmi, N. A., Fathani, A. T., Sadayi, D. P., Fitriani, I., & Rayhan Adiyaksa, M. (2021). Social Media Network Analysis (SNA): Identifikasi Komunikasi dan Penyebaran Informasi Melalui Media Sosial Twitter. *Jurnal Media Informatika Budidarma*, *5*(4), 1422–1430. https://doi.org/10.30865/mib.v5i4.3257

Daffa, W., Bamasag, O., & Almansour, A. (2018). A Survey on Spam URLs Detection in Twitter. *1st International Conference on Computer Applications and Information Security, ICCAIS 2018*, 1–6. https://doi.org/10.1109/CAIS.2018.8441975

Hary Candana, E. W., Gede, I., Gunadi, A., & Divayana, D. G. H. (2021). Perbandingan Fuzzy Tsukamoto, Mamdini Dan Sugeno Dalam Penentuan Hari Baik Pernikahan Berdasarkan Wariga Menggunakan Confusion Matrix. *Jurnal Ilmu Komputer Indonesia (JIK)*, *6*(2), 14–22.

Jatmiko, Y. A., Padmadisastra, S., & Chadidjah, A. (2019). Analisis Perbandingan Kinerja Cart Konvensional, Bagging Dan Random Forest Pada Klasifikasi Objek: Hasil Dari Dua Simulasi. *Media Statistika*, *12*(1), 1. https://doi.org/10.14710/medstat.12.1.1-12

Rahmi, I. A., Afendi, F. M., & Kurnia, A. (2023). Metode AdaBoost dan Random Forest untuk Prediksi Peserta JKN-KIS yang Menunggak. *Jambura Journal of Mathematics*, *5*(1), 83–94. https://doi.org/10.34312/jjom.v5i1.15869

Retnoningsih, E., & Pramudita, R. (2020). Mengenal Machine Learning Dengan Teknik Supervised Dan Unsupervised Learning Menggunakan Python. *Bina Insani Ict Journal*, *7*(2), 156. https://doi.org/10.51211/biict.v7i2.1422

Rezeki, S. R. I. (2020). Penggunaan sosial media twitter dalam komunikasi organisasi (studi kasus pemerintah provinsi dki jakarta dalam penanganan covid-19). *Journal of Islamic and Law Studies*, *04*(02), 63–78.

Rizaldi, D. F., Abdillah, J., Naufal, M., Yaqin, M. A., & Fauzan, A. C. (2022). Survei Pengukuran Fleksibilitas Software Menggunakan Metode Systematic Literature Review. *ILKOMNIKA: Journal of Computer Science and Applied Informatics*, *4*(1), 53–66. https://doi.org/10.28926/ilkomnika.v4i1.253

Ruth, D., Candraningrum, D. A., Tf-idf, M., Andriyani, N. A., Akhir, T., Pinandito, A., Setya Perdana, R., Chandra Kusuma, D. N. S., Oktavianti, R., Hasiholan, T. P., Pratami, R., Wahid, U., Kurniawan, H., Bayu Rahmawan, Tri Ginanjar Laksana, A. E. A., Utami, A. S. F., Baiti, N., Rosiyadi, D., Hidajat, M., Adam, A. R., … Rozana, A. N. (2019). Jurnal Sustainable : Jurnal Hasil Penelitian dan Industri Terapan Deteksi Twitter Bot Menggunakan Klasifikasi Decision Tree. *ComTech: Computer, Mathematics and Engineering Applications*, *15*(2), 257–262. http://ejournal.bsi.ac.id/ejurnal/index.php/cakrawala/article/view/3680/2624%0Ahttp://j-ptiik.ub.ac.id

Sandag, G. A. (2020). Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest. *CogITo Smart Journal*, *6*(2), 167–178. https://doi.org/10.31154/cogito.v6i2.270.167-178

Seno, D. W., & Wibowo, A. (2019). Analisis Sentimen Data Twitter Tentang Pasangan Capres-Cawapres Pemilu 2019 Dengan Metode Lexicon Based Dan Support Vector Machine. *Jurnal Ilmiah FIFO*, *11*(2), 144. https://doi.org/10.22441/fifo.2019.v11i2.004

Sobron, M., & Lubis. (2021). Implementasi Artificial Intelligence Pada System Manufaktur Terpadu. *Seminar Nasional Teknik (SEMNASTEK) UISU*, *4*(1), 1–7. https://jurnal.uisu.ac.id/index.php/semnastek/article/view/4134

Sodik, F., Dwi, B., & Kharisudin, I. (2020). Perbandingan Metode Klasifikasi Supervised Learning pada Data Bank Customers Menggunakan Python. *Jurnal Matematika*, *3*, 689–694.

Syamsiah, S. (2019). Perancangan Flowchart dan Pseudocode Pembelajaran Mengenal Angka dengan Animasi untuk Anak PAUD Rambutan. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, *4*(1), 86. https://doi.org/10.30998/string.v4i1.3623

Utomo, D. P. (2020). *Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung*. *4*(April), 437–444. https://doi.org/10.30865/mib.v4i2.2080

Wandani, A., Fauziah, F., & Andrianingsih, A. (2021). Sentimen Analisis Pengguna Twitter pada Event Flash Sale Menggunakan Algoritma K-NN, Random Forest, dan Naive Bayes. *J-SAKTI (Jurnal Sains Komputer Dan Informatika)*, *5*(2), 651–665.

Yusmita, A. R., Anra, H., & Novriando, H. (2020). Sistem Informasi Pelatihan pada Kantor Unit Pelaksana Teknis Latihan Kerja Industri (UPT LKI) Provinsi Kalimantan Barat. In *Jurnal Sistem dan Teknologi Informasi (Justin)* (Vol. 8, Issue 2, p. 160). https://doi.org/10.26418/justin.v8i2.36797

Zahra, A. A., Widyawan, W., & Fauziati, S. (2020). Development of Bot Detection Applications on Twitter Social Media Using Machine Learning with a Random Forest Classifier Algorithm. *IJITEE (International Journal of Information Technology and Electrical Engineering)*, *4*(2), 66. https://doi.org/10.22146/ijitee.56154