

MENGENAL

MACHINE LEARNING

ANALISIS SENTIMEN
MENGGUNAKAN NAIVE BAYES
CLASSIFIER



Dani Ferdinand
Arykan Anisa Pertiwi
Nisa Hanum Hariani

Mengenal *Machine learning*

Membangun Model Analisis Sentimen Menggunakan Naïve Bayes Classifier dan
Sistem *Webservice* Menggunakan Golang

**Dani Ferdinand
Aryka Anisa Pertiwi
Nisa Hanum Harani**



PT. Penerbit Buku Pedia
2023

Mengenal Machine learning

Membangun Model Analisis Sentimen Menggunakan Naïve Bayes Classifier dan Sistem Webservice Menggunakan Golang

Penulis:

Dani Ferdinand
Aryka Anisa Pertiwi
Nisa Hanum Harani

ISBN:

Editor:

-

Penyunting:

-

Desain sampul dan Tata letak:

Dani Ferdinand

Font:

Calibri

Penerbit:

PT. Penerbit Buku Pedia

Redaksi:

Athena Residence Blok. E No. 1, Desa Ciwaruga,
Kec. Parongpong, Kab. Bandung Barat 40559
Tel. 628-775-2000-300
Email : penerbit@bukupedia.co.id

Distributor:

Informatics Research Center
Jl. Sariasih No. 54
Bandung 40151
Email : irc@ulbi.ac.id

Cetakan Pertama, 2023

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara apa pun tanpa ijin tertulis dari penerbit

PRAKATA

Pada tahun 2023, penggunaan *machine learning* (ML) telah semakin meluas di berbagai bidang. Hal ini didorong oleh kemajuan teknologi yang semakin pesat, serta ketersediaan data yang semakin berlimpah.

Seiring dengan meningkatnya penggunaan ML, kebutuhan akan tenaga kerja yang kompeten di bidang ini juga semakin meningkat. Oleh karena itu, penting bagi kita untuk mempelajari ML, baik secara teori maupun praktik.

Buku ini hadir untuk memenuhi kebutuhan tersebut. Buku ini membahas tentang dasar-dasar ML, serta cara membangun sistem *webservice* menggunakan ML dengan bahasa pemrograman GoLang. Tujuan penulisan buku ini adalah untuk memberikan pemahaman dasar tentang ML, menjelaskan cara membangun sistem *webservice* menggunakan ML, dan membantu pembaca untuk mengembangkan keterampilan ML.

Buku ini ditujukan bagi pembaca yang ingin mempelajari ML, baik secara teori maupun praktik. Pembaca sasaran buku ini adalah mahasiswa, profesional, dan siapa saja yang tertarik dengan ML.

Buku ini memiliki beberapa keunggulan, antara lain pembahasan yang lengkap dan mendalam, contoh-contoh yang mudah dipahami, langkah-langkah yang jelas dan terstruktur, serta kode-kode yang dapat diakses melalui GitHub. Kode-kode dalam buku ini dapat diakses melalui GitHub, dan URL GitHub untuk masing-masing bab dapat ditemukan di akhir bab. Kode-kode dalam GitHub telah dikelompokkan berdasarkan bab dan dilengkapi dengan penjelasan yang cukup lengkap.

Selain itu, buku ini juga membahas implementasi *webservice* yang *deploy* melalui Google Cloud Function. *Webservice* ini menjadi penghubung antara model *machine learning* yang dibangun menggunakan Python dan *backend* yang menggunakan GoLang untuk mengolah *database* serta memanggil model Naïve Bayes Classifier (NBC). Penggunaan Google Cloud Function mempermudah pemanggilan fungsi-fungsi pada *backend* melalui API (*application programming interface*), sehingga integrasi antara *frontend* dan *backend* menjadi lebih efisien.

Studi dari *Fortune Business Insights* menyoroti pertumbuhan yang signifikan dalam pasar machine learning baru-baru ini. Pada tahun 2021, nilai pasar machine learning mencapai \$15,44 miliar USD. Proyeksi untuk akhir tahun 2022 memperkirakan peningkatan menjadi \$21,17 miliar USD, dan ekspektasi jangka panjang menyatakan bahwa pada tahun 2029, nilai pasar machine learning akan mencapai \$209,91 miliar USD. Trend ini mencerminkan adopsi yang semakin luas dari teknologi machine learning di berbagai sektor, menunjukkan potensi besar yang diakui oleh berbagai industri untuk memanfaatkan kecerdasan mesin dalam meningkatkan efisiensi dan inovasi. Data tersebut menunjukkan bahwa ML memiliki potensi yang sangat besar untuk digunakan dalam berbagai bidang. Oleh karena itu, penting bagi kita untuk mempelajari ML.

Buku ini sendiri merupakan wujud dari kebutuhan akan pentingnya meningkatkan skill dengan tantangan yang akan dihadapi oleh perusahaan dan generasi milenial di Era Industri 4.0, terutama yang berkaitan dengan komitmen kerja/bisnis dan *adversity quotient* (AQ).

Buku ini disajikan dalam setiap bab yang berisi langkah-langkah untuk pembaca pemula. Setiap bab dibuka dengan definisi dan kerangka kerja. Pada bagian selanjutnya diterangkan cara membuat kode program beserta penjelasan setiap baris kode program tersebut. Pada bagian akhir disertakan kode program untuk latihan dan evaluasi pembaca. Setiap chapter atau bab yang disajikan dalam buku ini disertakan juga kode program yang bisa diakses melalui link dibawah ini.

https://github.com/bukped/bukpedp3_Mengenal_Machine_Learning

DAFTAR ISI

PRAKATA	i
DAFTAR ISI	iii
BAB 1 PEDAHLUAN	1
A. PENDAHULUAN.....	1
B. CAPAIAN PEMBELAJARAN	2
BAB 2 TEKNOLOGI YANG DIGUNAKAN.....	3
A. PYTHON	3
B. JUPYTER NOTEBOOK.....	4
C. VS CODE.....	5
D. POSTMAN	5
E. GITHUB	7
F. GOOGLE CLOUD PLATFORM.....	7
G. MONGODB.....	8
H. GOLANG	9
I. JAVASCRIPT.....	9
BAB 3 INSTALASI APLIKASI YANG DIGUNAKAN.....	11
A. INSTALASI JUPYTER NOTEBOOK	11
B. VS CODE.....	13
C. POSTMAN	16
D. MONGODB.....	18
E. GOLANG	20
BAB 4 DASAR DASAR ANALISIS SENTIMEN DAN PREPOCESSING	23
A. PENGERTIAN ANALISIS SENTIMEN.....	23
B. SUMBER DATA SENTIMEN	25

C. IMPLEMENTASI	28
BAB 5 NAIVE BAYES UNTUK ANALISIS SENTIMEN.....	46
A. KONSEP DASAR NAÏVE BAYES.....	46
B. IMPLEMENTASI MENGGUNAKAN PYTHON	48
BAB 6 MEMBANGUN WEBSERVICE DENGAN GOLANG.....	54
A. PENGENALAN GOLANG.....	54
B. PEMBUATAN PACKAGE.....	55
C. UPLOAD BACKEND MENJADI PACKAGE.....	57
BAB 7 INTEGRASI DENGAN CLOUD FUNCTIONS.....	59
A. PENGANTAR CLOUD FUNCTIONS	59
B. IMPLEMENTASI INTEGRASI CLOUD FUNCTIONS PADA BACKEND	60
C. IMPLEMENTASI INTEGRASI CLOUD FUNCTIONS PADA MODEL ML.....	64
BAB 8 MEMBANGUN FRONTEND.....	73
A. PENDAHULUAN.....	73
B. IMPEMENTASI.....	74
BAB 9 MENGUNGGAH JS MENCADI CDN.....	84
A. PENDAHULUAN.....	84
B. IMPLEMENTASI	85
BAB 10 KESIMPULAN	88
A. KESIMPULAN.....	88
B. SARAN	89
DAFTAR PUSTAKA.....	91
GLOSARIUM	97
KREDIT GAMBAR	102
TENTANG PENULIS	103

BAB 1

PEDAHULUAN

A. PENDAHULUAN

Dalam era digital yang terus berkembang, akses mudah ke layanan internet telah meresapi kehidupan sehari-hari masyarakat Indonesia. Perkembangan ini tidak hanya menciptakan kemudahan akses, tetapi juga mengubah paradigma interaksi sosial melalui media sosial seperti Twitter, Instagram, dan Facebook. Dalam konteks ini, masyarakat tidak hanya konsumen informasi, tetapi juga aktor yang berpartisipasi aktif, menggambarkan pandangan dan perasaan mereka melalui unggahan gambar dan komentar (Atmodjo et al., 2014; Hasan et al., 2023; Ishaq & Mahanani, 2018; Noorikhsan et al., 2023; Wulan Ayu & Fitriyanto, 2022).

Pentingnya akses mudah ke internet dan penggunaan media sosial dalam kehidupan sehari-hari telah mengilhami penelitian dan eksplorasi lebih lanjut mengenai pemahaman sentimen masyarakat. Data yang berasal dari komentar-komentar di media sosial dapat menjadi sumber daya yang berharga untuk penelitian berbasis sentimen, mencakup berbagai topik seperti sentimen Masyarakat terhadap isu yang sedang hangat saat ini yaitu pemilihan presiden 2024. Studi kasus ini akan membuka peluang untuk melihat bagaimana masyarakat merespons peristiwa politik yang krusial melalui media sosial (Delvika et al., 2023; Fais Sya' bani et al., 2022; Jimmy et al., 2023; Pasek et al., 2022).

Seiring dengan perkembangan teknologi, pendekatan yang melibatkan *webservice* dan *cloud computing* semakin mendapatkan perhatian. Studi kasus pemilihan presiden 2024 akan dianalisis dengan menggunakan *machine learning* dengan algoritma pemodelan Naïve Bayes Classifier yang dibangun dengan bahasa pemrograman Python. Python dipilih untuk pengembangan model *machine learning* karena python merupakan Bahasa yang popular digunakan dalam pengembangan model *machine learning* (Buntoro et al., 2021; Koneru, Bhavani, et al., 2018).

Untuk menampilkan hasil dari pemodelan, *frontend* dibangun menggunakan HTML dengan framwork css Bulma dan Javascript standar Es6.

Untuk kebersihan kode Javascript akan diunggah menjadi CDN dengan Jsdelivr. Selain memberikan wawasan terhadap sentimen masyarakat, tutorial ini juga akan memberikan panduan praktis dalam mengimplementasikan teknologi-teknologi tersebut (Ardiansyah, 2019; Manullang et al., 2023; Puad & Susilo Yuda Irawan, 2023; Yusuf et al., 2020).

Lalu untuk menghubungkan model yang dibangun menggunakan Python, dibangun juga *Backend* menggunakan Golang untuk mengola *database* dan memanggil model NBC yang akan menganalisis data. Untuk mempermudah penggunaan *backend* oleh *frontend*, dibangunlah *webservice* menggunakan Google Cloud Function yang akan memanggil *package backend* dan menjadikannya *Api (application programming interface)* sehingga mempermudah pemanggilan fungsi fungsi pada *backend* (Biswas et al., 2022; Hadian et al., 2023; Lusita et al., 2022).

Dengan pendekatan yang berbasis pada kasus pemilihan presiden 2024, serta penggunaan teknologi *webservice*, *cloud computing*, dan *frontend development*, buku tutorial ini bertujuan memberikan pemahaman mendalam kepada pembaca tentang analisis sentimen yang dapat diterapkan pada berbagai konteks dalam era digital ini.

B. CAPAIAN PEMBELAJARAN

Dalam perancangan buku tutorial ini, tujuan instruksional menjadi fondasi utama untuk memandu proses pembelajaran.

1. Memahami prinsip-prinsip dan cara membangun *machine learning*
2. Membangun sistem *webservice* menggunakan golang
3. Mengintegrasikan *backend* dengan google cloud functions
4. Menggunakan API yang dibangun pada *frontend*

BAB 2

TEKNOLOGI YANG DIGUNAKAN

A. PYTHON



Gambar 2.1 Logo Python

Python, sebuah bahasa pemrograman tingkat tinggi, dirancang oleh Guido Van Rossum dan diperkenalkan pada tahun 1991. Saat ini, Python menikmati popularitas yang tinggi sebagai bahasa pemrograman serbaguna. Keunggulan Python terletak pada kemampuannya untuk mendukung berbagai kebutuhan, termasuk pengembangan *Machine learning* dan *Deep Learning* (Rifandi et al., 2021).

Sintaks Python terkenal karena kemudahannya, membuatnya dapat diakses oleh berbagai tingkat pengguna. Selain itu, kekayaan *library* Python dan dukungan yang kuat dari komunitas pengembangnya memperkuat daya tariknya. Sebagai perangkat lunak sumber terbuka, Python mengundang kontribusi dan kolaborasi dari berbagai pihak (Mahrozi & Faisal, 2023).

Dalam penulisan source code Python, pengembang dapat memilih berbagai *Integrated Development Environment* (IDE) seperti VS Code, Sublime Text, atau PyCharm. Selain itu, terdapat opsi penggunaan IDE *online* seperti Jupyter Notebook dan Google Colab. Python mampu mencakup berbagai bidang penggunaan, mulai dari pengembangan aplikasi web dan desktop hingga *Internet of Things* (IoT) dan beragam aplikasi lainnya.

Ada beberapa alasan mengapa Python menjadi pilihan utama, yaitu:

1. Python dapat berjalan di berbagai platform seperti Windows, Linux, macOS, Android, Raspberry Pi, dan lain-lain.
2. Python memiliki sintaks yang sederhana dan mirip dengan bahasa Inggris.

3. Sintaks Python memungkinkan penulisan kode yang lebih ringkas dibandingkan dengan bahasa pemrograman lain.
4. Python menggunakan interpreter, sehingga program dapat dieksekusi dengan cepat setelah selesai dibuat.
5. Python mendukung paradigma pemrograman prosedural, berorientasi objek, dan fungsional.

B. JUPYTER NOTEBOOK



Gambar 2.2 Logo Jupyter

Jupyter Notebook adalah aplikasi berbasis web yang dapat digunakan untuk membuat dan membagikan dokumen yang berisi kode, persamaan matematika, visualisasi, dan teks. Jupyter Notebook dikelola oleh projek Jupyter, sebuah proyek sumber terbuka yang berfokus pada pengembangan perangkat lunak, standar, dan layanan untuk komputasi interaktif dalam berbagai bahasa pemrograman. Jupyter Notebook merupakan singkatan dari tiga bahasa pemrograman: Julia, Python, dan R. Ketiga bahasa pemrograman ini penting bagi data scientist karena sering digunakan untuk analisis data, *machine learning*, dan *artificial intelligence* (Lavandaia Dharma Bali et al., 2021).

C. VS CODE



Gambar 2.3 Logo Visual Studio Code

Visual Studio Code (VSCode) merupakan perangkat lunak penyunting kode sumber yang dikembangkan oleh Microsoft untuk sistem operasi Linux, macOS, dan Windows. Fungsinya mencakup berbagai fitur penting seperti penyorotan sintaksis, penyelesaian kode, kutipan kode, refaktorisasi kode, pengawakutuan, dan integrasi dengan Git. Visual Studio Code dirilis oleh Microsoft dengan kode sumber terbuka di repositori GitHub, yang dilisensikan dengan lisensi MIT (Code - OSS) (Hafidah & Frendiana, 2023).

Meskipun kode sumbernya tersedia dengan lisensi MIT, perlu dicatat bahwa biner yang dibangun oleh Microsoft untuk Visual Studio Code tidak dirilis dengan lisensi MIT. Sebagai gantinya, versi biner ini bersifat berpemilik. Dengan demikian, sumber kode terbuka dan dapat diakses oleh komunitas pengembang, namun penggunaan versi biner tertentu dapat tunduk pada ketentuan lisensi khusus yang diterapkan oleh Microsoft.

D. POSTMAN



Gambar 2.4 Logo Postman

Postman adalah alat pengembangan API yang komprehensif yang dapat digunakan untuk mengembangkan, menguji, dan mendokumentasikan API (Arshad Busro Cahyono et al., 2022).

Fitur-fitur Postman meliputi:

1. Pembuatan permintaan HTTP yang mudah: Postman memungkinkan pengembang untuk membuat permintaan HTTP menggunakan berbagai metode, seperti *GET*, *POST*, *PUT*, dan *DELETE*.

2. Pengelolaan variabel yang fleksibel: Postman menyediakan fitur variabel yang memudahkan pengelolaan nilai yang dapat digunakan secara dinamis dalam permintaan API.
3. Pengujian API yang komprehensif: Postman mendukung pengujian API manual dan otomatis, serta menyediakan berbagai fitur untuk membantu pengembang membuat skenario pengujian kompleks.
4. Dokumentasi API yang otomatis: Postman dapat digunakan untuk membuat dokumentasi otomatis dari koleksi permintaan API, sehingga memudahkan berbagi informasi dengan tim atau komunitas.
5. Kolaborasi yang mudah: Postman memungkinkan pengembang untuk berbagi koleksi dan lingkungan, sehingga memudahkan kolaborasi dalam pengembangan API.
6. Keamanan yang terjamin: Postman menggunakan enkripsi HTTPS untuk memastikan keamanan data yang dikirimkan melalui API.
7. Postman adalah alat yang berharga bagi pengembang API karena dapat membantu mereka mengembangkan, menguji, dan mendokumentasikan API dengan lebih mudah dan efisien.

Berikut adalah beberapa contoh bagaimana Postman dapat digunakan:

1. Pengembang dapat menggunakan Postman untuk membuat permintaan HTTP ke API yang mereka buat untuk menguji fungsionalitas API.
2. Pengembang dapat menggunakan Postman untuk membuat skenario pengujian otomatis untuk API mereka.
3. Pengembang dapat menggunakan Postman untuk membuat dokumentasi API yang mudah dipahami oleh pengguna.
4. Tim pengembang dapat menggunakan Postman untuk berbagi koleksi permintaan dan lingkungan, sehingga memudahkan kolaborasi dalam pengembangan API.

Postman adalah alat yang serbaguna yang dapat digunakan untuk berbagai keperluan pengembangan API. Dengan antarmuka yang ramah pengguna dan fitur-fitur yang komprehensif, Postman dapat membantu pengembang API meningkatkan produktivitas dan kualitas pekerjaan mereka.

E. GITHUB



Gambar 2.5 Logo Github

GitHub adalah platform layanan hosting web bersama yang dirancang untuk proyek pengembangan perangkat lunak. Platform ini menggunakan sistem kendali versi Git, yang memungkinkan pengembang bekerja secara kolaboratif pada kode sumber proyek. GitHub menyediakan berbagai fitur untuk mendukung kolaborasi, termasuk kontrol akses, pelacakan bug, permintaan fitur, manajemen tugas, dan wiki untuk setiap proyek yang dihosting (Ferdiana Sari, 2021).

Penggunaan GitHub sangat umum di kalangan pengembang perangkat lunak karena memberikan infrastruktur yang kuat untuk mengelola kode sumber secara efisien dan bekerja sama dengan tim. Platform ini memungkinkan pengguna untuk menyimpan, melacak, dan mengelola versi perubahan dalam proyek perangkat lunak.

GitHub pertama kali dikembangkan pada tanggal 19 Oktober 2007, dan situs ini resmi diluncurkan pada April 2008 oleh Tom Preston-Werner, Chris Wanstrath, dan PJ Hyett. Sejak itu, GitHub telah menjadi salah satu platform terkemuka untuk kolaborasi pengembangan perangkat lunak dan menyediakan tempat untuk ribuan proyek *open source* dan swasta.

F. GOOGLE CLOUD PLATFORM



Gambar 2.6 Logo Google Cloud Platform

Google Cloud Platform (GCP) merupakan rangkaian layanan komputasi awan yang disediakan oleh Google. GCP dioperasikan di atas infrastruktur yang sama yang digunakan oleh produk internal Google seperti Google Search, YouTube, dan Gmail. Dengan dukungan alat manajemen yang komprehensif, GCP menawarkan berbagai layanan cloud modular yang mencakup komputasi, penyimpanan data, analisis data, dan pembelajaran mesin (Nur et al., 2023).

Layanan-layanan tersebut memungkinkan pengguna untuk membangun, mengelola, dan men-deploy aplikasi dan layanan dengan menggunakan infrastruktur yang dikelola oleh Google di pusat data mereka. Hal ini memberikan fleksibilitas dan skalabilitas kepada organisasi atau pengembang yang menggunakan GCP (Fernando, 2018).

Penting untuk dicatat bahwa untuk menggunakan layanan GCP, pengguna perlu mendaftar dan menyediakan detail kartu kredit atau rekening bank. Ini adalah langkah umum dalam proses pendaftaran untuk membantu mengelola biaya dan pembayaran layanan yang digunakan. Dengan demikian, GCP memberikan solusi yang luas dan terintegrasi untuk kebutuhan komputasi awan dan layanan terkait bagi berbagai jenis pengguna, dari perusahaan besar hingga pengembang individu.

G. MONGODB



Gambar 2.7 Logo Mongo DB

MongoDB adalah sebuah sistem basis data yang mengadopsi model dokumen dan termasuk dalam kelompok sistem basis data NoSQL. NoSQL, singkatan dari "Not Only SQL," menunjukkan bahwa sistem basis data tersebut tidak terbatas pada penggunaan bahasa kueri SQL untuk memanipulasi data. Berbeda dengan basis data relasional, MongoDB tidak bergantung pada struktur tabel, kolom, dan baris. Sebaliknya, dalam MongoDB, data disimpan dalam bentuk koleksi dan dokumen (Cahyo Santoso et al., 2020).

Dokumen dalam MongoDB dapat dianggap sebagai entitas yang menyimpan data. Yang menarik, dokumen dalam satu koleksi bisa memiliki struktur atribut yang berbeda satu sama lain, tidak seperti di basis data relasional yang membutuhkan struktur yang konsisten di setiap baris. Ini memungkinkan fleksibilitas yang lebih besar dalam menangani data yang memiliki format yang beragam atau evolusi skema yang cepat. Jadi, MongoDB memungkinkan penyimpanan dan manipulasi data tanpa memerlukan struktur yang ketat, memberikan keleluasaan dalam mengelola informasi yang bervariasi dan berubah seiring waktu (Septia Maharani et al., 2022).

H. GOLANG



Gambar 2.8 Logo Golang

Golang, atau disebut juga *Go Language*, adalah bahasa pemrograman yang dirancang untuk menggabungkan keamanan dan kinerja tinggi dalam pengembangan sistem. Bahasa ini bersifat *open source* dan dikembangkan di Google oleh tim pengembang yang terdiri dari Rob Pike, Robert Griesemer, Ken Thompson, dan berbagai kontributor lainnya dalam komunitas pengembang *open source* (Huang et al., 2019).

Golang dirancang dengan fokus pada kesederhanaan, efisiensi, dan kecepatan kompilasi. Tujuan utamanya adalah menyediakan lingkungan pengembangan yang efisien dan mudah dipahami, terutama untuk pembangunan sistem dan perangkat lunak berskala besar. Golang telah mendapatkan popularitas yang signifikan dalam beberapa tahun terakhir dan mulai banyak digunakan di perusahaan-perusahaan besar maupun *startup* di berbagai bidang teknologi.

Keunggulan Golang meliputi kinerja tinggi, penanganan konkurenси yang efisien, pengelolaan memori yang baik, dan kemampuan untuk mengembangkan perangkat lunak yang mudah di-maintain. Golang juga memiliki lingkungan pengembangan yang kuat dan komunitas yang aktif, yang berkontribusi pada pertumbuhan adopsi bahasa ini dalam industri pengembangan perangkat lunak.

I. JAVASCRIPT



Gambar 2.9 Logo Javascript

JavaScript adalah sebuah bahasa pemrograman yang dikhawasukan untuk membuat halaman web menjadi lebih dinamis dan interaktif. Nama "JavaScript" mungkin menimbulkan kesan bahwa bahasa ini terkait dengan

Java, tetapi sebenarnya keduanya adalah bahasa yang berbeda (Park et al., 2017).

JavaScript berjalan pada sisi klien (client-side), yang berarti kode program dieksekusi di dalam browser pengguna. Fungsinya adalah memberikan kemampuan tambahan pada halaman web, menjadikannya lebih dinamis daripada halaman web statis yang hanya menggunakan HTML. JavaScript memungkinkan pengembang untuk mengontrol perilaku halaman web, merespons interaksi pengguna, dan memanipulasi elemen HTML secara dinamis.

Perbedaan utama antara HTML, yang digunakan untuk membuat struktur halaman web, dan JavaScript, yang digunakan untuk menambahkan interaktivitas, memberikan pengalaman pengguna yang lebih kaya. JavaScript memiliki dukungan kuat di berbagai browser, membuatnya menjadi pilihan yang umum digunakan dalam pengembangan web.

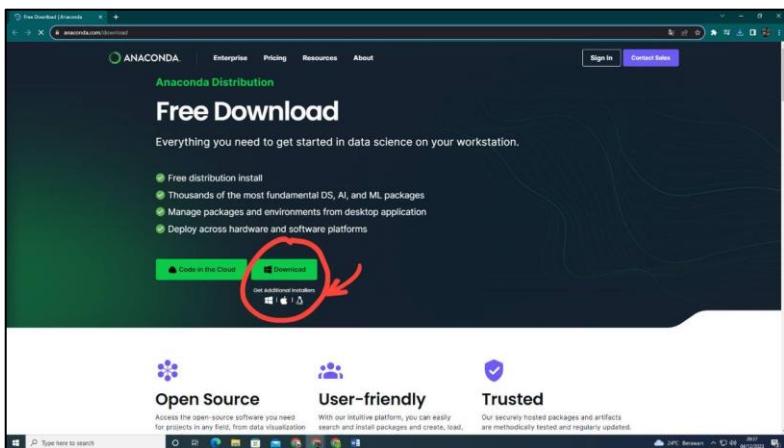
BAB 3

INSTALASI APLIKASI YANG DIGUNAKAN

A. INSTALASI JUPYTER NOTEBOOK

1. Mengunduh Anaconda

Silahkan kunjungi Halaman Unduhan Anaconda, setelah itu pilih versi yang sesuai dengan sistem operasi anda masing-masing (Windows, Linux, macOS).

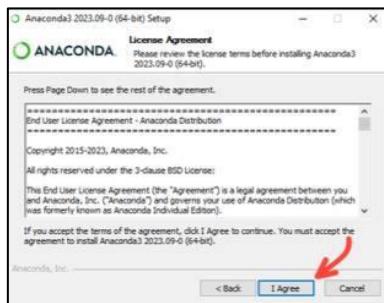


2. Instalasi Anaconda

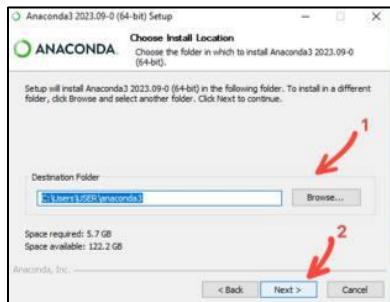
Setelah selesai melakukan pengunduhan, selanjutnya adalah buka file installer, ikuti petunjuk menginstall anaconda, jangan lupa sesuaikan dengan sistem operasi yang digunakan.

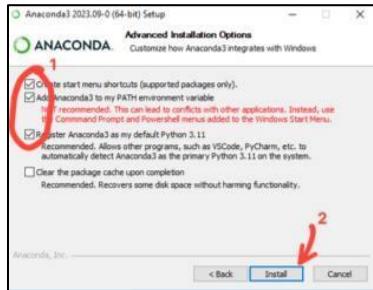


Klik *Next* lalu klik *I agree*.



Pada saat proses penginstalan pastikan kamu menceklist opsi “*add Anaconda to my PATH environment variable*” agar bisa menggunakan anaconda dari *command prompt/terminal* dengan mudah.

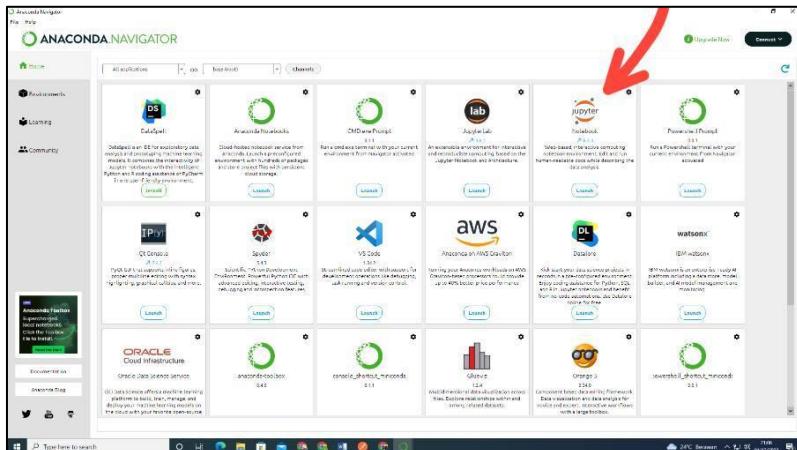




Terakhir Klik *Install* dan tunggu hingga proses selesai

3. Buka Jupyter Notebook

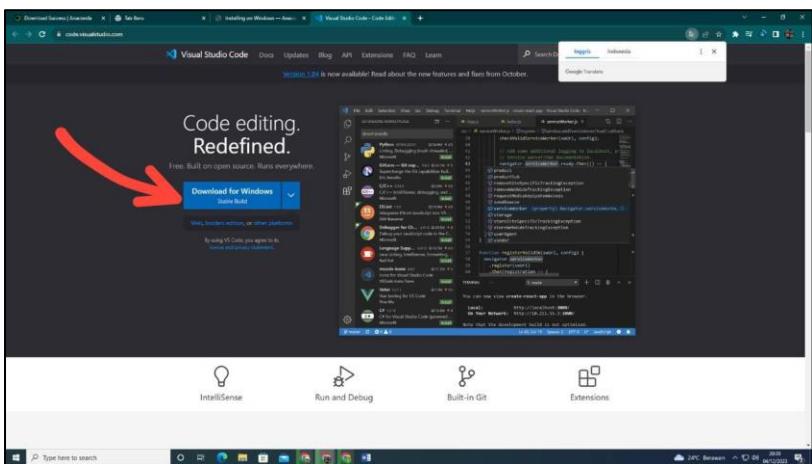
Setelah selesai melakukan instalasi buka anaconda melalui command prompt atau melalui terminal. Dengan begitu jupyter notebook sudah siap digunakan. Untuk informasi lebih lengkapnya anda bisa membaca di jupyter notebook documentation.



B. VS CODE

1. Mengunduh Visual Studio Code

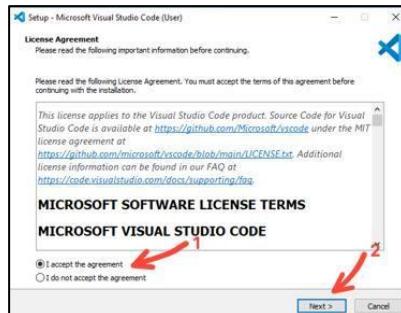
Untuk mengunduhnya silahkan kunjungi situs resmi visual studio code, lalu pilihlah sesuai dengan sistem operasi anda masing-masing.



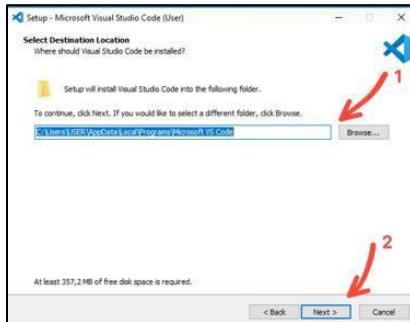
2. Instalasi Visual Studio Code

Setelah selesai mengunduh, buka file instalasi, lalu ikuti *wizard* instalasi, ubahlah jika perlu, dan mulai proses instalasi. Jika sudah maka visual studio code sudah siap digunakan.

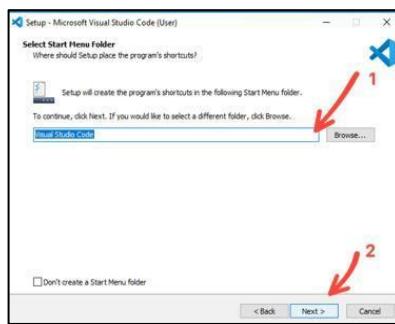
Untuk informasi lebih lanjut silahkan kunjungi Dokumentasi visual studio code. Anda juga dapat menambahkan ekstensi untuk mendukung Bahasa pemrograman yang akan kamu gunakan. Berikut Langkah-langkahnya.



Pilih *I accept the agreement* lalu klik *next*.



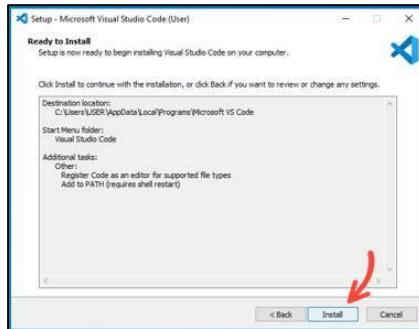
Pilih folder untuk instalasi Vs Code atau bisa juga menggunakan *default folder*, lalu klik *Next*.



Pilih folder untuk instalasi menyimpan *shortcut* Vs Code atau bisa juga menggunakan *default folder*, lalu klik *Next*.



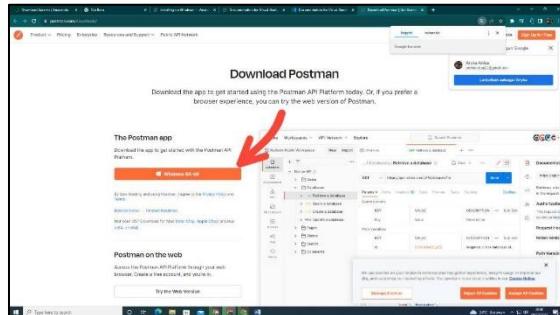
Terakhir klik *next* dan klik *install*, lalu tunggu hingga proses instalasi selesai.



C. POSTMAN

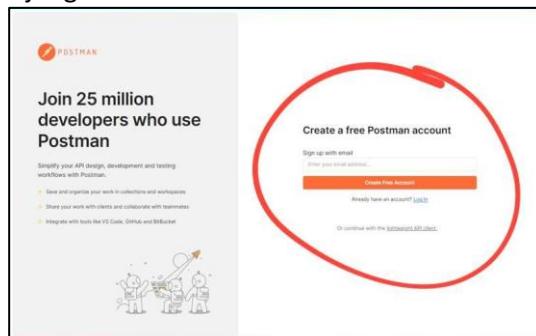
1. Mengunduh Postman

Silahkan kunjungi situs resmi postman untuk mengunduh postman. Jangan lupa pilihlah sesuai dengan sistem operasi yang anda gunakan.

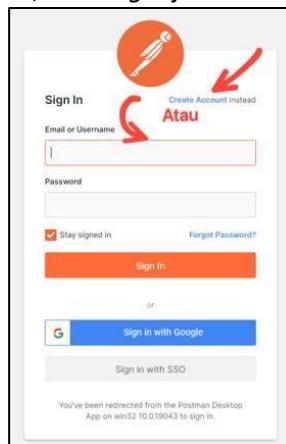


2. Instalasi Postman

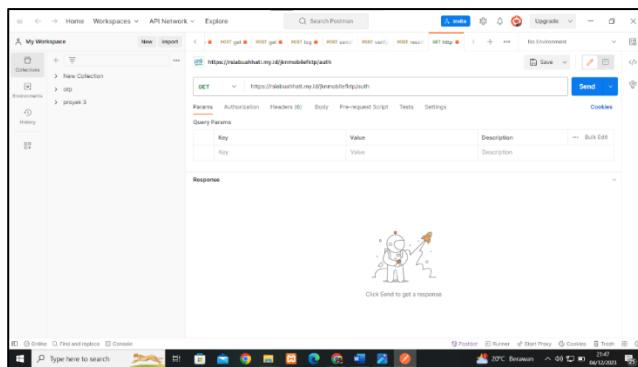
Buka file yang telah diunduh, lalu baca dengan teliti ikuti instruksi yang ditampilkan, lalu pilih sesuai dengan kebutuhan. Lakukan proses instalasi. Setelah selesai maka postman sudah siap digunakan. Untuk informasi lebih lanjut silahkan kunjungi dokumentasi Postman.



Buat akun untuk postman, atau *login* jika sudah punya.



Masukan email dan password, bisa juga menggunakan *google account*.

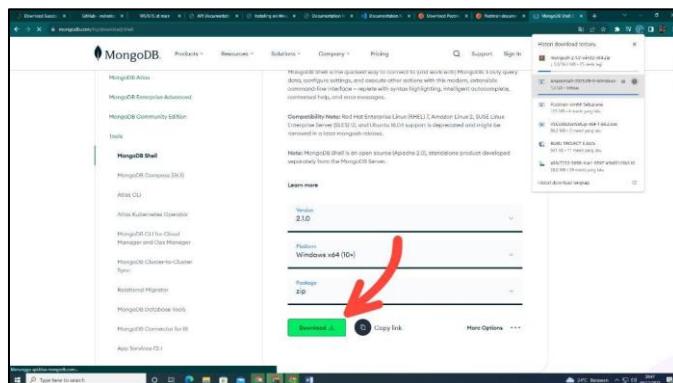


Lalu postman siap digunakan.

D. MONGODB

1. Mengunduh MongoDB

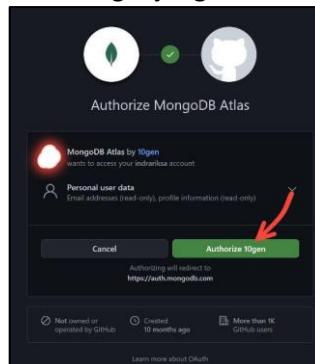
Untuk mengunduhnya silahkan untuk mengunjungi *Situs MongoDB Community Server*. Yang terpenting unduhlah sesuai dengan sistem operasi yang anda gunakan masing-masing.



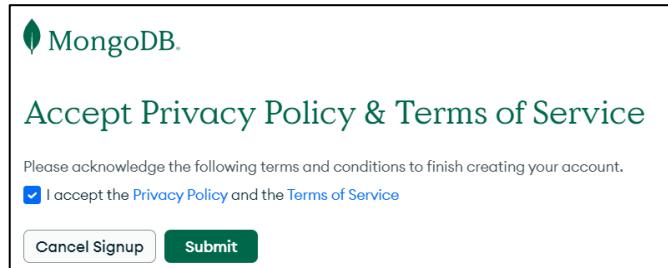
2. Instalasi MongoDB

Setelah mengunduh, selanjutnya adalah proses instalasi ikutilah instruksi yang ada, serta pilihlah opsi yang ada sesuai dengan kebutuhan anda masing-masing. Setelah menyelesaikan serangkaian proses instalasi, maka MongoDB sudah terinstall di komputermu.

Jika belum punya akun silahkan buat akun menggunakan akun Github (*sign in*) selanjutnya anda bisa mengunjungi Dokumentasi MongoDB.

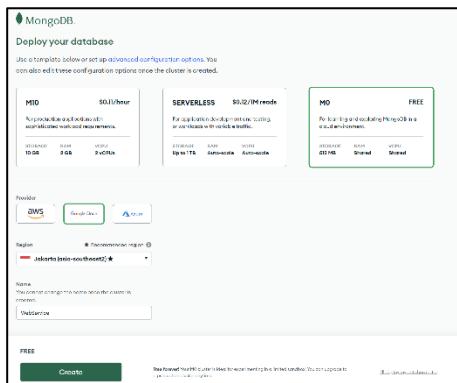


Klik **Authorize 10gen**



The screenshot shows a MongoDB account creation step. At the top is the MongoDB logo. Below it is the heading "Accept Privacy Policy & Terms of Service". A sub-instruction says "Please acknowledge the following terms and conditions to finish creating your account." There is a checked checkbox labeled "I accept the [Privacy Policy](#) and the [Terms of Service](#)". At the bottom are two buttons: "Cancel Signup" and "Submit".

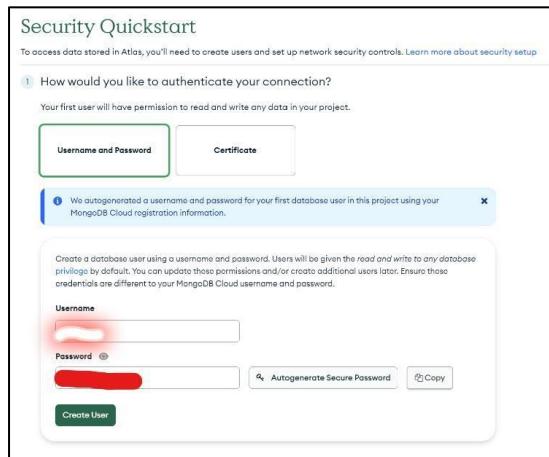
Klik *submit*



This screenshot shows the MongoDB deployment interface. It features three plan options: M10 (Standard), SERVERLESS (\$0.02/0M reads), and M0 (Free). Each plan has specific configuration details like storage and network settings. Below the plans are sections for "Provider" (AWS, Google Cloud, Azure), "Region" (set to "us-east-1"), and "Name" (set to "quickstart"). A "Create" button is at the bottom.

Klik *Create*.

Pada security *quickstart* silahkan pilih *username* dan *password* (Jangan lupa untuk save *password* di notepad kalian), kemudian *create user*.

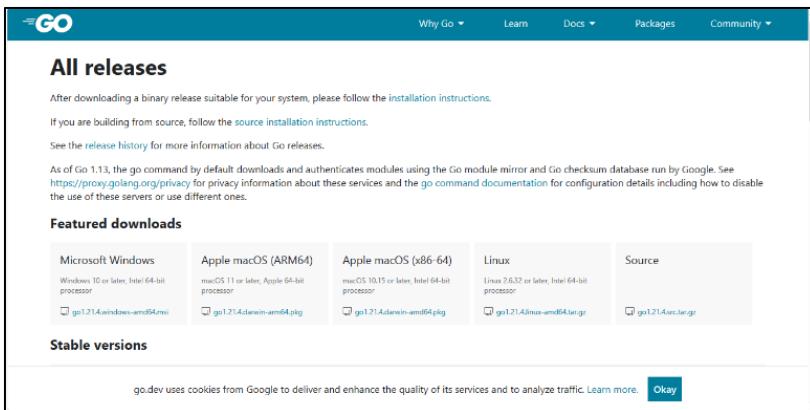


This screenshot shows the MongoDB security setup. It starts with a question "How would you like to authenticate your connection?", with "Username and Password" selected. A note says "We autogenerated a username and password for your first database user in this project using your MongoDB Cloud registration information." Below is a section for creating a database user with fields for "Username" and "Password". It also includes "Autogenerate Secure Password" and "Copy" buttons, and a "Create User" button at the bottom.

E. GOLANG

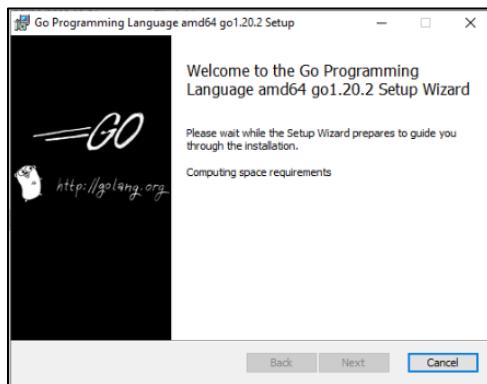
1. Mengunduh Golang

Unduhlah golang pada halaman resmo Go (Golang) di <https://go.dev/dl/>, unduhlah sesuai dengan sistem operasi dan arsitektur computer anda.

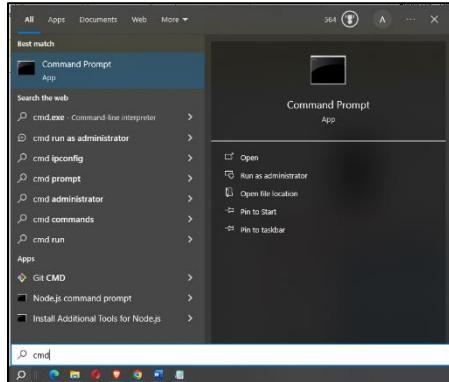


2. Instalasi Golang

Lakukan instalasi file yang telah diunduh dan ikutilah instruksi yang ada untuk memulai proses instalasinya.



Setelah instalasi selesai selanjutnya mengecek instalasi. Untuk melakukan pengecekan buka cmd pada menu start.



Ketikan go version pada cmd

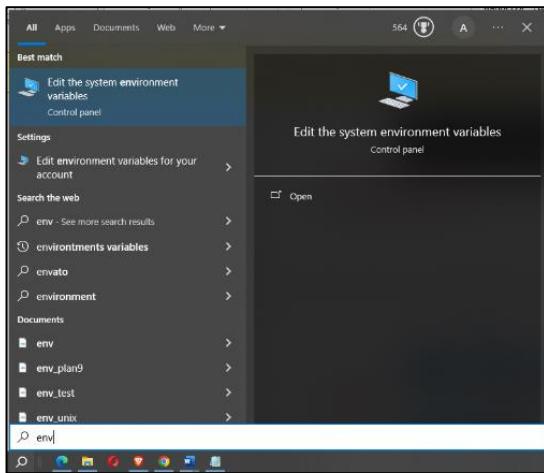
```
Command Prompt
Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\Users\aulia>go version
go version go1.21.2 windows/amd64

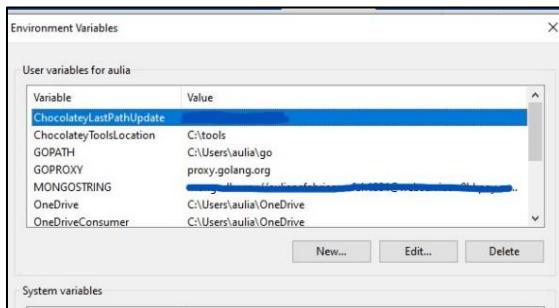
C:\Users\aulia>
```

A screenshot of a Command Prompt window titled 'Command Prompt'. The window shows the standard Microsoft Windows command line interface. At the top, it displays the path 'C:\Users\aulia>' followed by the command 'go version'. The output of the command, 'go version go1.21.2 windows/amd64', is shown below the command line. The window has a dark theme.

Jika sudah selanjutnya lakukan setup workspace/gopath. Untuk melakukan pengecekan buka *Edit the System Environment Variables* pada menu *start*.



Cari variabel GOPATH dan edit sesuai dengan direktori sesuai kebutuhan.



Jika sudah diedit cek status variable GOPATH dengan perintah go env. Jika selesai maka Golang sudah terinstal pada computer anda. Untuk informasi lebih bisa anda akses pada Dokumentasi Resmi Golang.

BAB 4

DASAR DASAR ANALISIS SENTIMEN DAN PREPOCESSING

A. PENGERTIAN ANALISIS SENTIMEN

1. Definisi dan ruang lingkup analisis sentimen.

Analisis sentimen adalah proses mengumpulkan, mengelola, dan mengevaluasi opini atau perasaan yang terkandung dalam teks, percakapan, atau data lainnya. Tujuan utamanya adalah untuk memahami dan mengevaluasi sentimen, sikap, atau opini yang mungkin terkandung dalam suatu konteks (Wahyu Sholeha et al., 2022). Beberapa elemen kunci dalam definisi ini mencakup:

- a. **Sentimen:** Reaksi emosional atau evaluatif terhadap suatu topik atau entitas.
- b. **Analisis:** Proses pemrosesan dan interpretasi data untuk mengidentifikasi dan memahami sentimen.
- c. **Teks dan Data:** Sumber informasi bisa berupa teks, percakapan *online*, ulasan produk, dan lainnya.

Ruang lingkup analisis sentimen melibatkan:

- **Pengumpulan Data:** Mengumpulkan data teks dari berbagai sumber.
- **Preprocessing:** Membersihkan dan memformat data untuk analisis.
- **Ekstraksi Fitur:** Identifikasi fitur atau kata-kata kunci yang berkaitan dengan sentimen.
- **Analisis Sentimen:** Menggunakan metode analisis untuk menentukan sentimen (positif, negatif, atau netral).
- **Visualisasi dan Pelaporan:** Menyajikan hasil analisis dengan cara yang mudah dimengerti

2. Aplikasi analisis sentimen dalam berbagai industri.

Analisis sentimen memiliki beragam aplikasi di berbagai industri, membantu organisasi untuk memahami pandangan pelanggan, mencari masalah potensial, atau memonitor reputasi (Christina & Ronaldo, 2020). Beberapa contoh industri dan aplikasinya melibatkan:

a. **Industri Perdagangan Elektronik (E-Commerce):**

- Melacak dan mengevaluasi ulasan produk.
- Mengukur kepuasan pelanggan.
- Menemukan trend dan preferensi konsumen.

b. **Industri Finansial:**

- Memantau opini dan sentimen pasar.
- Menganalisis umpan balik pelanggan terhadap layanan perbankan atau produk keuangan.

c. **Industri Kesehatan:**

- Menilai umpan balik pasien terhadap layanan medis.
- Mengidentifikasi trend kesehatan masyarakat melalui analisis data sosial media.

d. **Industri Pariwisata:**

- Mengevaluasi ulasan hotel, restoran, atau destinasi wisata.
- Mengukur citra dan reputasi destinasi wisata.

e. **Industri Teknologi:**

- Memantau sentimen terhadap produk dan layanan teknologi.
- Mengidentifikasi isu atau bug melalui umpan balik pengguna.

f. **Industri Media Sosial:**

- Menganalisis sentimen pengguna terkait berita, tren, atau peristiwa viral.
- Mendeteksi dan merespons krisis reputasi secara cepat.

g. **Industri Layanan Pelanggan:**

- Mengelola dan memperbaiki reputasi perusahaan melalui penanganan umpan balik pelanggan.
- Mengukur keefektifan kampanye pemasaran dan promosi.

Analisis sentimen dapat memberikan wawasan berharga bagi perusahaan, membantu mereka mengambil keputusan yang lebih baik berdasarkan pemahaman yang lebih baik tentang persepsi dan sentimen pelanggan atau pemangku kepentingan lainnya (Tineges et al., 2020).

B. SUMBER DATA SENTIMEN

1. Memilih Sumber Data Yang Sesuai Untuk Analisis Sentimen

Analisis sentimen merupakan proses kritis dalam pemahaman opini, perasaan, dan tanggapan masyarakat terhadap suatu topik atau entitas. Pemilihan sumber data yang tepat adalah langkah awal yang krusial dalam memastikan keberhasilan analisis sentimen. Sumber data yang sesuai akan memberikan wawasan yang lebih akurat dan relevan terkait dengan sentimen yang sedang dievaluasi. Dibawah ini adalah cara Memilih Sumber Data yang Sesuai:

a. Identifikasi Tujuan Analisis Sentimen:

Sebelum memilih sumber data, jelaskan mengenai tujuan analisis sentimen yang ingin dicapai. Apakah itu untuk memahami pandangan pelanggan, mengukur kepuasan konsumen, atau memonitor reputasi merk?

b. Pertimbangkan Industri atau Konteks:

Setiap industri memiliki sumber data yang lebih relevan. Misalnya, ulasan produk sangat penting untuk *e-commerce*, sementara diskusi forum mungkin lebih relevan untuk industri teknologi.

c. Lihat Keberlanjutan dan Volume Data:

Pilih sumber data yang menyediakan volume data yang memadai dan memastikan bahwa opini atau sentimen yang diungkapkan bersifat berlanjut, bukan hanya reaksi jangka pendek.

d. Perhatikan Keaslian dan Kredibilitas:

Pastikan bahwa sumber data adalah autentik dan kredibel. Media sosial dan situs ulasan yang terkenal dapat memberikan data yang lebih dapat diandalkan.

e. Filtering dan Preprocessing:

Pertimbangkan kemampuan untuk melakukan filtering dan preprocessing data. Sumber data yang memungkinkan pemrosesan awal akan membantu mengatasi noise atau informasi yang tidak relevan.

f. Perbandingan Sumber Data:

Bandingkan beberapa sumber data sebelum membuat keputusan akhir. Tinjau keberagaman opini, kecepatan pembaruan, dan kemudahan akses data.

g. Ketahui Keterbatasan Sumber Data:

Setiap sumber data memiliki keterbatasan. Kenali keterbatasan tersebut dan pertimbangkan apakah dapat diatasi atau diminimalkan.

h. Selaraskan dengan Model Analisis Sentimen yang Digunakan:

Pastikan bahwa sumber data yang dipilih sesuai dengan model analisis sentimen yang akan digunakan, apakah itu berbasis aturan, *machine learning*, atau kombinasi keduanya.

i. Perhatikan Privasi dan Kebijakan Data:

Pastikan bahwa penggunaan sumber data mematuhi kebijakan privasi dan hukum terkait. Privasi pengguna harus dihormati.

j. Evaluasi Secara Berkala:

Lakukan evaluasi rutin terhadap sumber data yang digunakan. Apakah masih sesuai dengan tujuan dan kebutuhan analisis sentimen yang sedang dilakukan.

2. Metode Pengumpulan Data yang Efektif (Crisp DM)

Untuk pengumpulan data sentimen yang efektif, langkah-langkah yang terstruktur dan terorganisir secara sistematis sangat penting. Oleh karena itu digunakan metode pendekatan CRISP-DM (*Cross Industry Standard Process for Data Mining*) (Singgalen, 2023b). Berikut adalah langkah-langkah untuk pengumpulan data sentimen yang efektif:

a. Pemahaman Bisnis (*Business Understanding*):

- Tujuan: Definisikan tujuan analisis sentimen dan identifikasi kebutuhan bisnis.
- Tindakan: Rancang pertanyaan atau tujuan spesifik terkait sentimen yang perlu dijawab.

b. Pemahaman Data (*Data Understanding*):

- Tujuan: Memahami karakteristik dan kualitas data yang tersedia.
- Tindakan: Eksplorasi sumber data potensial, identifikasi pola, dan cek kualitas data.

- c. Persiapan Data (*Data Preparation*):
 - Tujuan: Menyiapkan data untuk analisis sentimen.
 - Tindakan: Membersihkan data dari *noise*, mengelola missing values, dan konversi format jika diperlukan.
- d. Modeling:
 - Tujuan: Menggunakan model atau algoritma untuk menganalisis sentimen.
 - Tindakan: Pilih model analisis sentimen yang sesuai, seperti algoritma *machine learning* atau teknik pemrosesan bahasa alami (NLP).
- e. Evaluasi:
 - Tujuan: Menilai kinerja model dan hasil analisis sentimen.
 - Tindakan: Gunakan metrik evaluasi yang sesuai, seperti akurasi, presisi, dan recall.
- f. Implementasi:
 - Tujuan: Mengimplementasikan hasil analisis sentimen ke dalam pengambilan keputusan bisnis.
 - Tindakan: Buat laporan atau dashboard yang dapat dimengerti oleh pemangku kepentingan.
- g. Pemeliharaan (*Deployment and Maintenance*):
 - Tujuan: Menjaga keakuratan dan relevansi model analisis sentimen.
 - Tindakan: Perbarui model secara berkala, sesuaikan dengan perubahan tren atau kebutuhan bisnis.

Dengan memadukan sumber data yang sesuai dengan kebutuhan dan mengikuti metodologi CRISP-DM, perusahaan dapat mengoptimalkan proses pengumpulan data sentimen dan meningkatkan keakuratan analisis mereka (Singgalen, 2023a).

C. IMPLEMENTASI

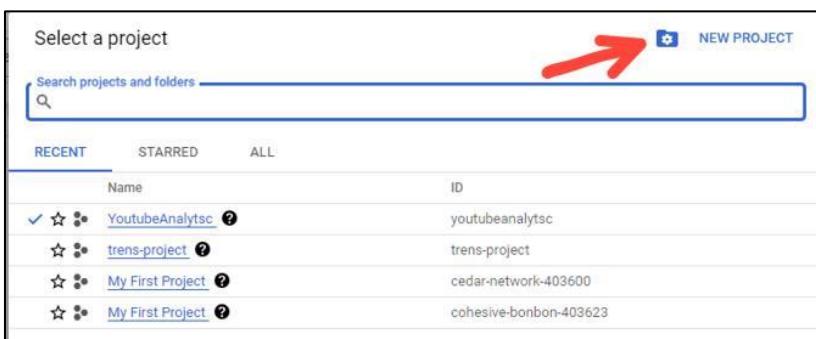
1. Crawling Atau Pengumpulan Data

Untuk melakukan *Crawling* data komentar youtube, kita membutuhkan data API youtube. Untuk mendapatkan api tersebut, pertama kita buat projek google cloud platform.

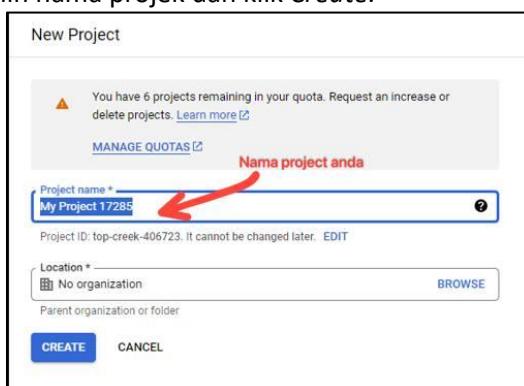
Mendapatkan Apikey;

a. Lakukan login atau registrasi google cloud platform

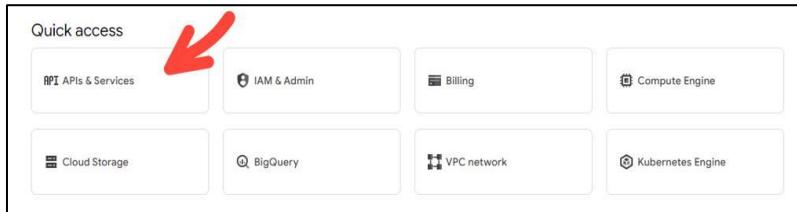
b. buat *New Project*



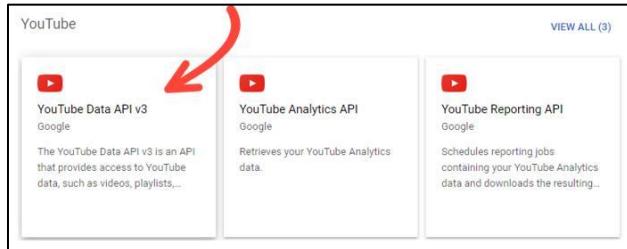
Setelah itu pilih nama projek dan klik *Create*.



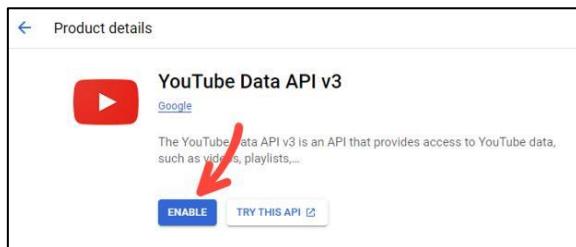
c. Jika sudah berhasil *create new project*, masuk dan cari *Quick Access*. kemudian pilih *APIs & Services*



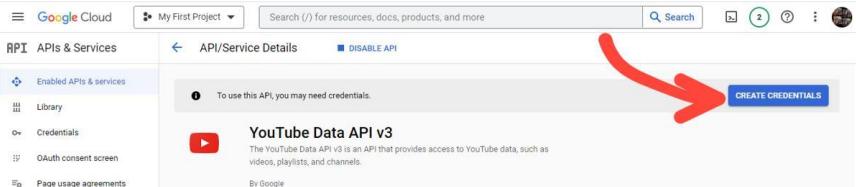
- d. Scrool ke bawah hingga menemukan Youtube seperti dibawah, kemudian pilih Youtube Data API v3



- e. Pilih *ENABLE*



- f. Lakukan *Create credentials*, ikuti prosesnya, serta isilah apa yang diminta sesuai dengan data kamu.



g. Pilih Next.

The screenshot shows the 'Create credentials' page. A red arrow labeled '1' points to the 'Select an API' dropdown menu, which is set to 'YouTube Data API v3'. Another red arrow labeled '2' points to the 'Public data' radio button, which is selected. A third red arrow labeled '3' points to the 'NEXT' button at the bottom.

h. Api Key diatas yang nantinya akan anda gunakan dalam proses Crawling data. jangan lupa Klik Tombol Done.

The screenshot shows the 'Your Credentials' page. It displays the 'Your API key' section. A red arrow labeled '4' points to the 'RESTRICT KEY' button. Below it, a red box highlights the 'API Key' field, which contains a redacted value, and a red arrow points to the trash icon next to it.

- i. Jika kredensial yang kamu buat sudah benar maka akan muncul pada *APIs & SERVICES* seperti dibawah ini. Klik pada *enabled APIs & Services* lalu nanti ada pada daftar.

The screenshot shows the Google Cloud Platform interface for managing APIs. On the left, there's a sidebar with options like 'Enabled APIs & services', 'Library', 'Credentials', 'OAuth consent screen', and 'Page usage agreements'. A red arrow labeled '1' points to the 'Enabled APIs & services' link. The main pane lists various Google APIs. A red arrow labeled '2' points to the 'YouTube Data API v3' entry in the list.

Double klik pada kredensial yang kamu buat maka akan diarahkan ke halaman API Keys. lalu bisa pilih *show Key* untuk melihat API Key kamu. API ini yang nanti akan digunakan dalam proses *Crawling* data komentar youtube

The screenshot shows the 'API/Service Details' page for the 'YouTube Data API v3'. It displays the 'API Keys' section, which lists a single credential named 'API key 1'. A red arrow points to the 'SHOW KEY' button next to the credential's row.

Proses *Crawling* Data

- a. Mengimport Library dan Koneksi API

```
import pandas as pd
from googleapiclient.discovery import build
```

- b. Mengambil komentar pada youtube, sesuai dengan link yang dipilih
Kita buat fungsi untuk mendapatkan data youtube dengan nama video_comments.

```
def video_comments(video_id):
```

- c. Lalu buatlah *list* kosong, guna menyimpan data komentar serta balasannya

```
# empty list for storing reply
replies = []
```

- d. Membuat youtube resource objek

```
# creating youtube resource object
youtube = build('youtube', 'v3', developerKey=api_key)
```

- e. Mengambil komentar serta balasan dengan video_id yang menggunakan list() serta mengembalikan hasilnya kedalam bentuk video_response.

```
# retrieve youtube video results
video_response = youtube.commentThreads().list(part='snippet,replies', videoId=video_id).execute()
```

- f. Berguna untuk mengolah *response* pada video.

```
# iterate video response
while video_response:
```

- g. Berguna untuk menjelajahi terhadap semua komentar serta balasan yang ada pada hasil *response* dari panggilan ke API youtube.

```
#for item in video_response['items']:
```

- h. Kode berikut berisi informasi mengenai waktu publikasi komentar serta nama penggunanya.

```
# Extracting comments ()
published = item['snippet']['topLevelComment']['snippet']['publishedAt']
user = item['snippet']['topLevelComment']['snippet']['authorDisplayName']
```

- i. Kode berikut berfungsi untuk mengambil isi dari komentar dan berapa jumlah Sukanya dari setiap komentar yang terdapat pada *response* API youtube.

```
# Extracting comments
comment = item['snippet']['topLevelComment']['snippet']['textDisplay']
likeCount = item['snippet']['topLevelComment']['snippet']['likeCount']
```

- j. Selanjutnya adalah menambahkan info yang telah di ambil pada komentar ke dalam list dalam bentuk array.

```
replies.append([published, user, comment, likeCount])
```

- k. Berikut merupakan kode untuk mengambil jumlah total dari balasan yang terkait dengan suatu komentar dalam *response* API youtube.

```
replycount = item['snippet']['totalReplyCount']
```

- l. Memeriksa pada suatu komentar terdapat balasan atau tidak (tidak=0).

```
if replycount>0:
```

- m. Mengambil setiap balasan komentar yang berkaitan dengan komentar yang sedang diproses.

```
for reply in item['replies']['comments']:
    # Extract reply
    published = reply['snippet']['publishedAt']
    user = reply['snippet']['authorDisplayName']
    repl = reply['snippet']['textDisplay']
    likeCount = reply['snippet']['likeCount']
```

- n. Mengambil waktu publikasi balasan komentar, nama, isi dari balasan, serta jumlah suka.

```
#replies.append([published, user, repl, likeCount])
```

- o. Fungsi if else dimana if itu terjadi apabila terdapat komentar berikutnya maka permintaan baru akan di buat sebaliknya jika else atau tidak ada maka akan break.

```
# Again repeat
if 'nextPageToken' in video_response:
    video_response = youtube.commentThreads().list(
        part = 'snippet,replies',
        pageToken = video_response['nextPageToken'],
        videoId = video_id
    ).execute()
else:
    break
```

- p. Mengeluarkan replies sehingga anda dapat menggunakan datanya.

```
return replies
```

Berikut ini merupakan proses memfungksikan kode yang telah dideklarasikan pada kode sebelumnya. Lebih spesifiknya mengambil komentar youtube menggunakan API sesuai dengan video yang dipilih. Berikut kode yang di maksud:

- a. Memasukkan token API yang tadi di dapat dari GCF

```
# isikan dengan api key Anda
api_key = '*****'
```

- b. Isikan video id, sesuaikan dengan apa yang anda butuhkan

```
# Enter video id
# contoh url video =https://www.youtube.com/watch?v=8MYIRwboPoM
video_id = "pDVf0uEY73k" #isikan dengan kode / ID video
```

- c. Panggil fungsi berikut untuk mengambil hasil dari komentar serta balasannya dari video, sesuai dengan ID yang anda inputkan

```
# Call function  
comments = video_comments(video_id)  
  
comments
```

Selanjutnya adalah memfungsikan pandas, agar komen youtube yang sudah kita ambil tadi bisa lebih rapi dan mudah dimengerti, seperti berikut:

```
df = pd.DataFrame(comments, columns=['publishedAt', 'authorDisplayName', 'textDisplay', 'likeCount'])  
df
```

Yang terakhir adalah mengekspor dataframe ke dalam bentuk csv. Serta menghindari menyimpan index data frame ke dalam file csv.

```
df.to_csv('dataset.csv', index=False)
```

2. Membersihkan dan Merapikan Data Untuk Analisis Sentiment

```
pip install clean-text
```

Pertama kita *install* dulu *module* python yang diperlukan. Adalah alat bantu untuk membersihkan dan memproses teks secara umum. Ini menyediakan fungsi-fungsi untuk membersihkan dan memformat teks, seperti menghapus karakter yang tidak diinginkan, mengubah huruf menjadi huruf kecil, dan banyak lagi.

```
import pandas as pd
import numpy as np

import os

from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory
from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemover
from Sastrawi.StopWordRemover.StopWordRemoverFactory import ArrayDictionary

from tqdm import tqdm

import re
import string
from cleantext import clean
```

Selanjutnya *Import* semua *module* yang diperlukan. Berikut adalah penjelasan dari setiap *module*.

- a. pandas (pd) dan numpy (np): Modul untuk bekerja dengan struktur data tabular dan operasi numerik di Python. Pandas digunakan untuk manipulasi data dalam bentuk DataFrame, sementara NumPy
- b. os: Modul yang menyediakan fungsi-fungsi untuk berinteraksi dengan sistem operasi, seperti membaca atau menulis file, mengatur variabel lingkungan, dan lainnya.
- c. Sastrawi: Merupakan modul Bahasa Indonesia yang menyediakan alat untuk pemrosesan bahasa alami (NLP), termasuk penghapusan kata-kata stop (Stop Word Removal). Dalam kode ini, modul StopWordRemoverFactory dan ArrayDictionary dari Sastrawi digunakan untuk mengelola kata-kata stop.
- d. tqdm: Modul untuk menampilkan bar progres (progress bar) saat melakukan iterasi atau proses yang memerlukan waktu. Berguna untuk melihat perkembangan proses secara visual.
- e. re dan string: Modul re menyediakan operasi ekspresi reguler (regex) untuk pencarian dan manipulasi teks. Modul string menyediakan beberapa fungsi untuk memanipulasi string, seperti penghapusan karakter tertentu.
- f. cleantext: Paket yang telah diinstal dengan perintah pip install clean-text sebelumnya. Digunakan untuk membersihkan dan

memproses teks, termasuk penghapusan karakter non-alfanumerik.

```
# Specify the path to the directory containing your CSV files
folder_path = 'dataset/'

# Initialize an empty list to store DataFrames
dfs = []

# Iterate through each file in the specified directory
for filename in os.listdir(folder_path):
    if filename.endswith('.csv'):
        # Construct the full path to the CSV file
        file_path = os.path.join(folder_path, filename)

        # Read the CSV file into a DataFrame
        df = pd.read_csv(file_path)

        # Append the DataFrame to the list
        dfs.append(df)

# Concatenate all DataFrames in the list into a single DataFrame
combined_df = pd.concat(dfs, ignore_index=True)
```

Kode diatas merupakan kode untuk mengambil semua data dengan extensi csv dari folder **dataset**. Pada gambar dibawah adalah dataframe yang telah digabung.

combined_df				
	publishedAt	authorDisplayName	textDisplay	likeCount
0	2023-12-07T13:51:52Z	@ideesti1833	Semoga terwujud Aamiim presiden	0.0
1	2023-12-07T10:50:47Z	@smithwerben1058	MANTAPPPP	0.0
2	2023-12-07T05:37:35Z	@MahendraDatta-f1vz	Anies cocok jadi Presiden 8 Ganteng, amana...	0.0
3	2023-12-07T03:35:32Z	@CaturWidodo-lh2je	jujur demi Allah biarkan kalian mengoceh yang ...	0.0
4	2023-12-06T23:18:54Z	@asepsodikin6801	Sy dukung prabowo krn wakilnya bkan saatnya sk...	0.0
...
135179	2023-10-25T11:01:26Z	Gus Wedi	Woo...woo , ditipu berkali kali kok masih perc...	1.0
135180	2023-10-25T10:56:53Z	Singa Anom	Hoax lagi.....<a href="about:i...	0.0
135181	2023-10-25T10:56:12Z	Rakyat Jelata	Nggak salah kok pak, biar sama-sama nyungsep n...	3.0
135182	2023-10-25T11:29:48Z	Anak Terpelajar	Amin	0.0
135183	2023-10-25T10:55:30Z	karma berjalan	Ada anak yang kepingin wapres malah yang dapat...	0.0
135184 rows × 4 columns				

Lalu kita cek dataframe tersebut dengan fungsi info().

```
combined_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 135184 entries, 0 to 135183
Data columns (total 4 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   publishedAt      135184 non-null   object  
 1   authorDisplayName 135113 non-null   object  
 2   textDisplay       135083 non-null   object  
 3   likeCount         135052 non-null   float64 
dtypes: float64(1), object(3)
memory usage: 4.1+ MB
```

Gambar diatas menampilkan data kolom, tipe data dan jumlah data setiap kolom. Setelah itu kita buang kolom yang tidak diperlukan.

df = combined_df.drop(columns=['authorDisplayName','likeCount'])		
df		
	publishedAt	textDisplay
0	2023-12-07T13:51:52Z	Semoga terwujud Aamiim presiden
1	2023-12-07T10:50:47Z	MANTAPPPP
2	2023-12-07T05:37:35Z	Anies cocok jadi Presiden 8 Ganteng, amana...
3	2023-12-07T03:35:32Z	jujur demi Allah biarkan kalian mengocah yang ...
4	2023-12-06T23:18:54Z	Sy dukung prabowo krn wakilnya bkan saatnya sk...
...
135179	2023-10-25T11:01:26Z	Woo...woo , ditipu berkali kali kok masih perc...
135180	2023-10-25T10:56:53Z	Hoax lagi.....<a href="about:i...
135181	2023-10-25T10:56:12Z	Nggak salah kok pak, biar sama-sama nyungsep n...
135182	2023-10-25T11:29:48Z	Amin
135183	2023-10-25T10:55:30Z	Ada anak yang kepingin wapres malah yang dapat...
135184 rows × 2 columns		

Pada gambar diatas terlihat kolom yang dihapus adalah “authorDisplayName” dan “likeCount”. Setelah itu kita hitung data *null* atau data kosong seperti pada gambar dibawah.

```
df.isnull().sum()
```

```
publishedAt      0
textDisplay     101
dtype: int64
```

Pada gambar diatas Terlihat ada pada kolom “textDisplay” ada 101 data kosong. Lalu pada gambar dibawah kita hapus data kosong dengan fungsi dropna().

```
df_cleaned = df.dropna()

# Tampilkan informasi tentang DataFrame yang sudah dibersihkan
print("Info for Cleaned DataFrame:")
df_cleaned.isnull().sum()

Info for Cleaned DataFrame:

publishedAt      0
textDisplay      0
dtype: int64
```

Setelah data dipastikan tidak ada data kosong, kita looping data tersebut dan bersihkan satu persatu dan dimasukan kedalam *list* clean_text.

```

clean_texts = []

stop_factory = StopWordRemoverFactory().get_stop_words() #Load default stopword

for text in tqdm(df_cleaned['textDisplay']):
    text = re.sub("@[A-Za-z0-9]+","",text) #hapus @
    text = re.sub(r"<[^>]*", "", text) #hapus tag html
    text = re.sub("(?:@|http?://|https?://|www)\S+", "", text) #Remove http links
    text = text.replace("#", "").replace("_", " ") #Remove hashtag sign but keep the text
    text = re.sub(r"\d+", "", text) # Remove number
    text = text.replace('""','') #remove quotation mark
    text = text.lower() #Lower Case
    text = text.strip() # Remove Whitespace
    text = text.translate(str.maketrans("", "", string.punctuation)) #Remove Punctuation
    text = clean(text, no_emoji=True) # Remove emoji

    # Menghapus Stopword
    dictionary = ArrayDictionary(stop_factory)
    swr = StopWordRemover(dictionary)
    text = swr.remove(text)

    clean_texts.append(text)

```

Gambar diatas bisa dilihat bahwa;

- Data dibersihkan dari at (@)
- Data dibersihkan dari tag html
- Data dibersihkan dari link
- Data dibersihkan dari hashtag
- Data dibersihkan dari nomor
- Data dibersihkan dari tanda kutip dan petik
- Data ubah menjadi *lowercase*
- Data dibersihkan dari *Whitespace*
- Data dibersihkan dari tanda baca
- Data dibersihkan dari emoji

Setelah itu data *clear* dimasukkan kedalam dataframe.

```
df_cleaned.loc[:, 'clear'] = clean_texts
```

Setelah data bersih didapatkan, kita cek ulang apakah ada data kosong. Kita ubah dulu data string kosong menjadi data *Nan* atau *null*.

```
df_cleaned['clear'].replace('', np.nan, inplace=True) # replace string kosong dengan Nan
```

Lalu kita hitung lagi apakah ada data data yang kosong, dan dari gambar dibawah terlihat data kosong ada 2310. Setelah itu kita hupus data yang kosong tersebut.

```
df_cleaned.isnull().sum()

publishedAt      0
textDisplay      0
clear           2310
dtype: int64

# Hapus data kosong
df_cleaned = df_cleaned.dropna(subset=['clear'])

df_cleaned.isnull().sum()

publishedAt      0
textDisplay      0
clear           0
dtype: int64
```

Langkah terakhir adalah dataframe yang telah diubah kita export menjadi data excel untuk kolom **clear** dan **textDisplay** di export menjadi data csv. Data clear kita ubah menjadi excel karena kita akan *translate* data tersebut menggunakan bantuan dari google sheet pada point berikutnya.

```
df_cleaned[['clear']].to_excel('clear_dataset.xlsx', index=False)

df_cleaned[['textDisplay']].to_csv('raw_dataset.csv', index=False)
```

3. Memeri Labeling Pada Sentimen

Dikarenakan **textBlob** hanya mendukung Bahasa Inggris, kita akan menerjemahkan sentimen berbahasa Indonesia menjadi Bahasa Inggris. Untuk menerjemahkan kita menggunakan formula:

=GOOGLETRANSLATE(A1;"ID";"EN")

Setelah itu tekan kepala tabel **B**, dan tekan **Ctrl+D** untuk mengimplementasikan formula tersebut ke semua data.

=GOOGLETRANSLATE(A2;"ID";"EN")		
A	B	C
clear	translate	
semoga terwujud	Hopefully the Presidential Aamium will b	

Setelah itu unduh data tersebut dalam format csv.

```
pip install textblob
```

Kembali lagi ke kodingan, pertama kita *install module* textblob. Untuk klasifikasi sentimen.

```
import pandas as pd
import numpy as np

from tqdm import tqdm
from textblob import TextBlob
```

Selanjutnya kita import modul yang diperlukan, lalu kita import data csv yang telah di terjemahkan tadi dan data mentah yang belum di bersihkan.

df_translated = pd.read_csv('translated_dataset.csv')		
df_raw = pd.read_csv('raw_dataset.csv')		
df_translated		
0	semoga terwujud aamium presiden	Hopefully the Presidential Aamium will be real...
1	mantapppp	MANTAPPPP
2	anies cocok jadi presiden ganteng amanah pinta...	Anies is suitable to be a handsome president o...
3	jujur aliah biarkan kalian mengocok jadi presi...	Honestly Allah let you blabber in becoming pre...
4	sy dukung prabowo km wakilnya akan saatnya sk...	I support Prabowo because the representatives ...
...
132768	woowoo ditipu berkali kali kok percaya sil mikir	Woowoo was cheated many times why I believe in...
132769	hoax	Hoax
132770	salah kok pak biar samasama nyungsep nantinya	It's wrong, sir, so that you can work together...
132771	amin	Amen
132772	anak kepingin wapres malah dapat anak silent	Children want the vice president to get a sile...
132773 rows × 2 columns		

df_raw	
	textDisplay
0	Semoga terwujud Aamiim presiden
1	MANTAPPPP
2	Anies cocok jadi Presiden 8 Ganteng, amana...
3	jujur demi Allah biarkan kalian mengoceh yang ...
4	Sy dukung prabowa km wakilnya bkan saatnya sk...
...	...
132768	Woo...woo , ditipu berkali kali kok masih perc...
132769	Hoax lagi.....<a href="about:i...
132770	Nggak salah kok pak, biar sama-sama nyungsep n...
132771	Amin
132772	Ada anak yang kepingin wapres malah yang dapat...
132773 rows × 1 columns	

Setelah itu gabungkan kedua data tersebut seperti gambar dibawah.

df_combine = pd.concat([df_raw, df_translated], axis=1)			
	textDisplay	clear	translate
0	Semoga terwujud Aamiim presiden	semoga terwujud aamiim presiden	Hopefully the Presidential Aamiim will be real...
1	MANTAPPPP	mantapppp	MANTAPPPP
2	Anies cocok jadi Presiden 8-br>Ganteng, amana...	anies cocok jadi presiden ganteng amanan pinta...	Anies is suitable to be a handsome president o...
3	jujur demi Allah biarkan kalian mengoceh yang ...	jujur allah biarkan kalian mengoceh jadi presi...	Honestly Allah let you blabber in becoming pre...
4	Sy dukung prabowa km wakilnya bkan saatnya sk...	sy dukung prabowa km wakilnya bkan saatnya sk...	I support Prabowo because the representatives ...
...
132768	Woo...woo , ditipu berkali kali kok masih perc...	wooooo ditipu berkali kali kok percaya sih mikir	Woowoo was cheated many times why I believe in...
132769	Hoax lagi.....<a href="about:i...	hoax	Hoax
132770	Nggak salah kok pak, biar sama-sama nyungsep n...	salah kok pak biar samasama nyungsep nantinya	It's wrong, sir so that you can work together...
132771	Amin	amin	Amen
132772	Ada anak yang kepingin wapres malah yang dapat...	anak kepingin wapres malah dapat anak silent	Children want the vice president to get a sile...
132773 rows × 3 columns			

Kita cek ulang apakah ada data yang kosong pada kolom *translate*. Dan jika ada kita bersihkan.

```
: df_combine.isnull().sum()

: textDisplay      0
: clear           0
: translate       1
: dtype: int64
```

```
df_cleaned = df_combine.dropna()

# Tampilkan informasi tentang DataFrame yang sudah dibersihkan
print("Info for Cleaned DataFrame:")
df_cleaned.isnull().sum()

Info for Cleaned DataFrame:

textDisplay      0
clear           0
translate       0
dtype: int64
```

Setelah itu *looping* dataframe dan lakukan klasifikasi sentimen apakah sentimenter sebut positif, negatif atau netral dengan textblob seperti gambar dibawah. textblob mengembalikan dua nilai, *polarity* dan *subjectivity*. Untuk klasifikasi lakukan pemilihan sentimen berdasarkan nilai *polarity*, jika *polarity* dibawah 0 maka sentimen tersebut negatif, diatas 0 adalah positif dan 0 adalah netral. Kita tambah juga score untuk mendefinisikan nilai sentimen sebagai -1 untuk negatif, 0 untuk netral dan 1 untuk positif.

```
label_texts = []
polarity_text = []
subjectivity_texts = []
scores = []

for comment in tqdm(df_cleaned['translate']):
    try:
        blob = TextBlob(comment)
        sentimen = blob.sentiment

        # Mendapatkan nilai sentimen
        polarity = sentimen.polarity # Nilai antara -1 hingga 1 (positif hingga negatif)
        subjectivity = sentimen.subjectivity # Nilai antara 0 hingga 1 (faktual hingga subjektif)
    except Exception as e:
        print(f"Error at index {index}: {e}")
        print(comment)

        # Menampilkan hasil analisis sentimen
    if polarity > 0:
        sentiment_label = "Positif"
        score=1
    elif polarity < 0:
        sentiment_label = "Negatif"
        score=-1
    else:
        sentiment_label = "Netral"
        score=0

    scores.append(score)
    label_texts.append(sentiment_label)
    polarity_text.append(polarity)
    subjectivity_texts.append(subjectivity)
```

100% [██████████] 132772/132772 [00:14<00:00, 9269.43it/s]

Setelah itu masukan hasil textblob kedalam dataframe seperti gambar dibawah.

```
df_final = df_cleaned.copy()

df_final['label']=label_texts
df_final['score']=scores
df_final['polarity']=polarity_text
df_final['subjectivity']=subjectivity_texts
```

Lalu kita lihat dataframe yang telah di gabungkan.

df_final						
	textDisplay	clear	translate	label	score	polarity
0	Semoga terwujud Aamilm presiden	semoga terwujud aamilm presiden	Hopefully the Presidential Aamilm will be real...	Netral	0	0.000
1	MANTAPPPP	mantapppp	MANTAPPPP	Netral	0	0.000
2	Anies cocok jadi Presiden 8-bri-Ganteng amana...	anies cocok jadi presiden ganteng amanah pinta...	Anies is suitable to be a handsome president o...	Positif	1	0.525
3	Jujur demik Alih biarkan kalian mengoceh jadi yang ...	jujur alih biarkan kalian mengoceh jadi presi...	Honestly Allah let you blabber in becoming pre...	Positif	1	0.525
4	Sy dukung prabowo km wakilnya bkan saathya sk...	sy dukung prabowo km wakilnya bkan saathya sk...	I support Prabowo because the representatives ...	Positif	1	0.200
...
132768	Woo..woo , ditpu berkali kali kok masih perc...	woovooo ditpu berkali kali kok percaya siih mikir	Woovooo was cheated many times why I believe in...	Positif	1	0.500
132769	Hoax lagi.....	hoax	Hoax	Netral	0	0.000
132770	Nggak salah kok pak, bisa sama-sama nyungsep n...	salah kok pak biar samasama nyungsep nantinya	It's wrong, sir, so that you can work together...	Negatif	-1	-0.250
132771	Amin	amin	Amen	Netral	0	0.000
132772	Ada anak yang kepingin wapres malah yang dapat...	anak kepingin wapres malah dapat anak silent	Children want the vice president to get a sile...	Netral	0	0.10000

Terakhir export menjadi data csv untuk dilakukan pemodelan *machine learning* pada bab berikutnya.

```
: df_final.to_csv('labeling_dataset.csv', index=False)
```

BAB 5

NAIVE BAYES UNTUK ANALISIS SENTIMEN

A. KONSEP DASAR NAÏVE BAYES

Konsep dasar Naive Bayes dalam konteks analisis sentimen adalah dasar dari metode klasifikasi yang sangat efektif dalam memprediksi sentimen suatu teks atau dokumen. Metode ini didasarkan pada teorema probabilitas Bayes dan mengasumsikan kemandirian (naivitas) antar fitur-fitur yang digunakan dalam analisis (Niar et al., 2022).

1. Teorema Probabilitas Naïve Bayes

Naive Bayes memanfaatkan Teorema Probabilitas Bayes untuk menghitung probabilitas kelas tertentu berdasarkan fitur-fitur yang diamati. Rumus dasar teorema ini adalah:

$$P(\text{Kelas}|\text{Fitur}) = \frac{P(\text{Kelas}|\text{Fitur}) \times P(\text{Kelas})}{P(\text{Fitur})} \dots (1)$$

Dari rumus (1) dapat di lihat bahwa P merupakan probabilitas (kemungkinan), Kelas merupakan kejadian Kelas, dan Fitur merupakan kejadian Fitur. Rumus (1) digunakan untuk menghitung kemungkinan dari terjadinya Kelas ketika Fitur sedang terjadi (Khotimah & Utami, 2022).

2. Kemandirian (Naivitas)

Model Naive Bayes mengasumsikan bahwa setiap fitur yang digunakan dalam analisis sentimen adalah mandiri satu sama lain. Artinya, keberadaan atau nilai suatu fitur tidak dipengaruhi oleh keberadaan atau nilai fitur lainnya. Meskipun asumsi ini seringkali tidak realistik di dunia nyata, namun model ini tetap efektif dalam banyak kasus.

3. Probabilitas Fitur Terhadap Kelas

Naive Bayes menghitung probabilitas masing-masing fitur terhadap setiap kelas. Ini melibatkan pengamatan terhadap seberapa sering fitur-fitur tertentu muncul dalam kelas tertentu.

4. Model Klasifikasi Sentimen

Dalam konteks analisis sentimen, fitur-fitur dapat berupa kata-kata atau frasa tertentu yang dianggap penting. Model Naive Bayes kemudian menghitung probabilitas bahwa suatu teks atau dokumen termasuk dalam kelas sentimen positif, negatif, atau netral berdasarkan distribusi probabilitas fitur-fitur yang diamati.

5. Pemilihan Fitur Dan Pembobotan

Pemilihan fitur-fitur yang relevan dan pembobotan dapat mempengaruhi kinerja model. Beberapa metode seperti TF-IDF (Term Frequency-Inverse Document Frequency) dapat digunakan untuk memberikan bobot pada fitur-fitur yang lebih informatif.

6. Training Dan Prediksi

Model Naive Bayes dilatih dengan menggunakan dataset yang sudah diberi label sentimen. Setelah tahap pelatihan, model dapat digunakan untuk memprediksi sentimen dari teks baru berdasarkan probabilitas yang dihitung.

7. Manfaat dan Keterbatasan

Kelebihan Naive Bayes meliputi kinerja yang cepat dan efisien, terutama pada dataset besar. Namun, asumsi kemandirian dapat menjadi keterbatasan, terutama jika ada ketergantungan yang signifikan antar fitur-fitur.

8. Penerapan Pada Analisis Sentimen

Naive Bayes seringkali digunakan dalam analisis sentimen untuk mengklasifikasikan teks, ulasan, atau dokumen ke dalam kategori sentimen tertentu, seperti positif, negatif, atau netral.

B. IMPLEMENTASI MENGGUNAKAN PYTHON

Pertama kita import dulu modul modul yang diperlukan dalam melakukan pemodelan.

```
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
import matplotlib.pyplot as plt
import seaborn as sns
```

Pada gambar diatas, modul yang di import adalah;

- 1 pandas (pd) dan numpy (np): Kedua pustaka ini umumnya digunakan untuk manipulasi dan analisis data. pandas digunakan untuk mengelola data dalam struktur DataFrame, sedangkan numpy digunakan untuk operasi numerik.
- 2 scikit-learn (sklearn): Pustaka ini banyak digunakan untuk tugas machine learning. Dalam skrip ini, digunakan untuk pemilihan model (train_test_split), evaluasi kinerja (confusion_matrix, classification_report, accuracy_score), ekstraksi fitur teks (TfidfVectorizer, CountVectorizer), dan untuk mengimplementasikan klasifikasi Naive Bayes (MultinomialNB).
- 3 matplotlib dan seaborn: Kedua pustaka ini digunakan untuk visualisasi data. matplotlib.pyplot adalah pustaka untuk membuat grafik, sementara seaborn adalah pustaka visualisasi data statistik. Mereka sering digunakan bersama untuk membuat plot dan grafik yang menarik secara visual

Selanjutnya baca file dataset yang klasifikasi sentimennya.

```
df_labeling = pd.read_csv('labeling_dataset.csv')
```

	textDisplay	clear	translate	label	score	polarity	subjectivity
0	Semoga terwujud Aamilm presiden	semoga terwujud aamilim presiden	Hopefully the Presidential Aamilim will be real..	Netral	0	0.000	0.000000
1	MANTAPPPP	mantapppp	MANTAPPPP	Netral	0	0.000	0.000000
2	Anies cocok jadi Presiden 8 Ganteng, amana...	anies cocok jadi presiden ganteng amana pinta...	Anies is suitable to be a handsome president o...	Positif	1	0.525	0.725000
3	jujur demi Allah biarkan kalian mengoceh yang ...	jujur alih biarkan kalian mengoceh jadi presi...	Honestly Allah let you blabber in becoming pre...	Positif	1	0.525	0.875000
4	Sy dukung prabowa km wakilnya bkan Saatnya sk...	sy dukung prabowa km wakilnya bkan saatnya sk...	I support Prabowa because the representatives ...	Positif	1	0.200	0.366667
...
132767	Woo...woo .. ditipu berkali kali kok masih perc...	woowooo ditipu berkali kali kok percaya sih mikir	Woowoo was cheated many times why I believe in...	Positif	1	0.500	0.500000
132768	Hoax lagi.....<a href="#about.i...	hoax	Hoax	Netral	0	0.000	0.000000
132769	Nggak salah kok pak, biar sama-sama nyungsep n...	salah kok pak biar samasama nyungsep nantinya	It's wrong, sir, so that you can work together...	Negatif	-1	-0.250	0.450000
132770	Amin	amin	Amen	Netral	0	0.000	0.000000
132771	Ada anak yang kepingin wapres malah yang dapat...	anak kepingin wapres malah dapat anak silent	Children want the vice president to get a sile...	Netral	0	0.000	0.100000

132772 rows × 7 columns

Lalu kita pisahkan data `textDisplay` fitur atau data masukan yang disimpan pada variabel X, dan data score atau sentimen sebagai label.

```
X = df_labeling['textDisplay']
Y = df_labeling['score']
```

Setelah itu kita pisahkan data tersebut menjadi dua, yaitu data *train* dan data *test*, data *train* digunakan untuk melatih model *machine learning* dan data *test* digunakan untuk mengetes hasil pemodelan untuk mendapatkan akurasi.

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2,random_state=42)
```

Lalu karena data label berbentuk string, kita konversi string tersebut dengan `CountVectorizer` menjadi representasi vector, dapat digunakan pada pemodelan.

```
vectorizer = CountVectorizer()

tfidf_train_vectors = vectorizer.fit_transform(X_train)
tfidf_test_vectors = vectorizer.transform(X_test)
print("n_samples: %d, n_features: %d" % tfidf_train_vectors.shape)
```

Selanjutnya pada gambar dibawah, deklarasi model naïve bayes dan latih model tersebut menggunakan data *train*.

```
#model training and prediction

naive_bayes_classifier = MultinomialNB()
naive_bayes_classifier.fit(tfidf_train_vectors, Y_train)
```

Setelah selesai kita lakukan presiksi terhadap data test, dan gunakan clasification_report untuk mengevaluasi pemodelan.

```
Y_pred = naive_bayes_classifier.predict(tfidf_test_vectors)
```

```
# Tampilkan hasil
print(f'Accuracy: {accuracy}')
print('\nClassification Report:')
print(classification_report)

Accuracy: 0.7304085859536811

Classification Report:
      precision    recall  f1-score   support
       -1       0.61     0.30     0.41     2953
        0       0.80     0.72     0.76    11732
        1       0.69     0.85     0.76    11870

           accuracy                           0.73    26555
          macro avg       0.70     0.62     0.64    26555
    weighted avg       0.73     0.73     0.72    26555
```

Terlihat pada gambar diatas. Bahwa akurasi memiliki nilai 0.73 atau akurasinya adalah 73 persen.

Untuk mempercantik laporan akurasi kita tampilkan dengan grafik menggunakan kode seperti gambar[].

```

# ...

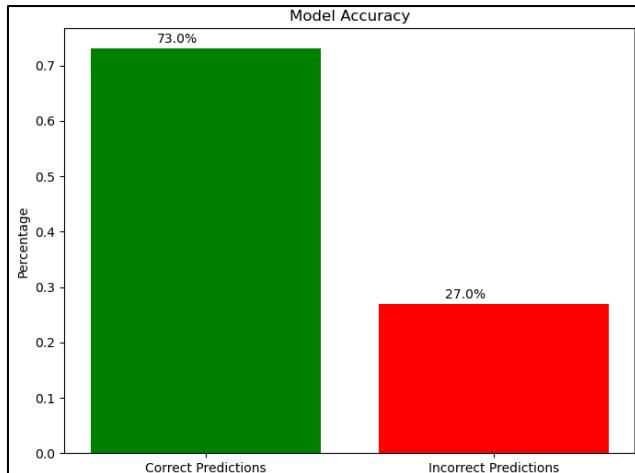
# Visualisasi Bar Chart untuk Akurasi Model dengan Keterangan Angka
accuracy_values = [accuracy, 1 - accuracy]
labels = ['Correct Predictions', 'Incorrect Predictions']

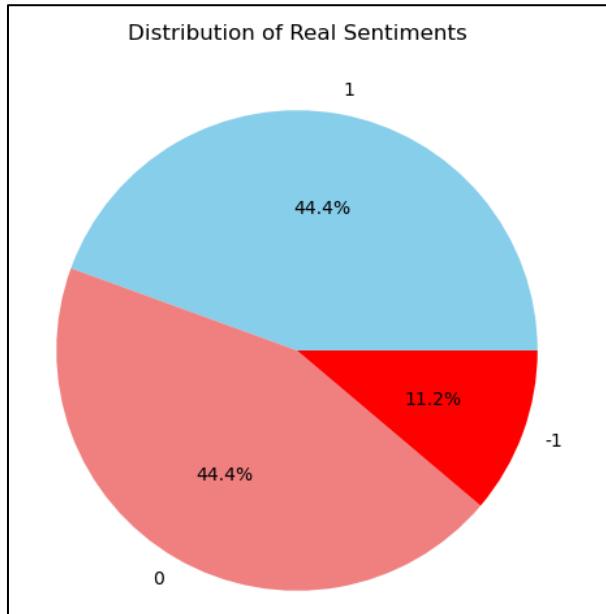
plt.figure(figsize=(8, 6))
bars = plt.bar(labels, accuracy_values, color=['green', 'red'])

# Tambahkan keterangan angka di atas setiap bar
for bar, value in zip(bars, accuracy_values):
    plt.text(bar.get_x() + bar.get_width() / 2 - 0.1, bar.get_height() + 0.01, f'{value*100:.1f}%', ha='center', color='black')

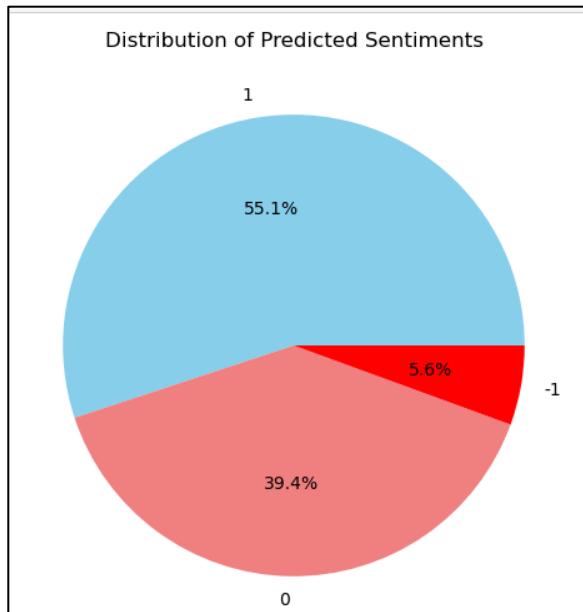
plt.title('Model Accuracy')
plt.ylabel('Percentage')
plt.show()

```





Gambar diatas merupakan distribusi sentimen pada prediksi data *test*.



Gambar diatas merupakan distribusi sentimen asli pada data *test*.

Lalu untuk menggunakan model tersebut, kita gunakan modul pickle untuk menyimpan model dalam bentuk file. kita simpan juga model dari vektorisasi nya untuk melakukan vektorisasi pada text yang dikirim dari *frontend*.

```
import pickle

with open('modelNB.pickle', 'wb') as file:
    pickle.dump(naive_bayes_classifier, file)

with open('vectorizer.pickle', 'wb') as file:
    pickle.dump(vectorizer, file)
```

BAB 6

MEMBANGUN WEBSERVICE DENGAN GOLANG

A. PENGENALAN GOLANG

Golang, atau disebut juga Go, adalah bahasa pemrograman open-source yang dikembangkan oleh Google. Dirancang dengan fokus pada produktivitas, kejelasan sintaksis, dan kinerja tinggi, Golang menjadi pilihan yang populer untuk pengembangan perangkat lunak skala besar dan aplikasi berkinerja tinggi (Simangunsong & Handoko, 2020).

1. Fitur Utama Golang

- Efisiensi Kompilasi: Golang menggunakan kompilasi statis, menghasilkan biner yang mandiri tanpa kebutuhan runtime eksternal.
- Manajemen Memori Otomatis: Pengelolaan memori otomatis dengan Garbage Collector, mengurangi beban pengembang.
- Sintaksis Jelas dan Konsisten: Sintaksis yang sederhana dan konsisten membuat kode mudah dibaca dan dipelajari.
- Paket Standar Kaya: Pustaka standar Golang mencakup fungsi-fungsi dasar yang dibutuhkan oleh sebagian besar aplikasi.
- Kinerja Tinggi: Cocok untuk pengembangan aplikasi berkinerja tinggi dan skala besar.

2. Aplikasi Golang yang Umum

- *Backend Web*: Golang sering digunakan untuk membangun *backend* web dengan kinerja tinggi.
- Mikroservis: Cocok untuk arsitektur mikroservis dengan dukungan konkurensi dan kinerja tinggi.
- Alat Sistem: Golang banyak digunakan untuk pengembangan alat sistem dan infrastruktur.

Dengan karakteristiknya yang unik, Golang menjadi pilihan yang menarik untuk membangun aplikasi dan layanan web yang efisien dan dapat diandalkan. Pada bagian selanjutnya, kita akan membahas langkah-langkah untuk membangun *webservice* menggunakan Golang.

B. PEMBUATAN PACKAGE

Pada bab ini, kita akan menyelami dunia *package* Go. Kita akan membahas mulai dari cara membuat *package*, memahami struktur internal, hingga menggunakannya secara efektif dalam program kita. Persiapkan diri untuk mempelajari teknik-teknik membangun aplikasi yang modular, scalable, dan efisien.

1. Membuat Projek

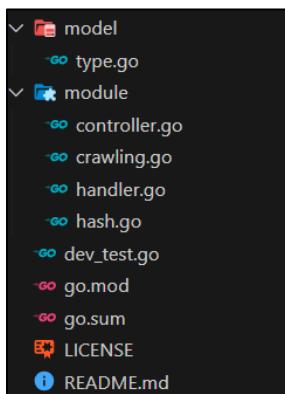


```
mkdir be_trensen
cd be_trensen
go mod init github.com/trentsentimen/be_trensen
```

Langkah pertama adalah membuat folder baru dengan fungsi **mkdir nama_projek** setelah itu masuk ke dalam folder yang baru dibuat dengan fungsi **cd nama_proyek**. Langkah terakhir adalah inisialisasi projek Golang dengan fungsi **go mod init nama_module**.

2. Struktur Projek

Setelah membuat projek baru sekarang kita akan membahas struktur projek yang akan dibangun. Dalam pembuatan *backend* ini kita akan membuat struktur yang baik, karena struktur proyek yang baik akan memastikan bahwa proyek dapat diskala dengan mudah dan dipelihara dengan baik.



Pada gambar diatas, fungsi fungsi file akan dijelaskan dibawah ini.

Model (type.go)

- a. type.go berisi definisi dari model atau struktur data yang digunakan dalam projek ini.
- b. type.go juga berisi dari type data dari setiap struktur ada.

Module (controller.go, Crawling.go, handler.go, hash.go)

- a. Pada direktori ini mencakup fungsi modul utama dalam projek, seperti fungsi fungsi utama, penanganan permintaan API dan hash untuk *password*.
- b. Controller.go berisi dari fungsi-fungsi yang merupakan logika utama, yang menghubungkan *handler* dengan model.
- c. Handler.go berisi fungsi untuk menerima permintaan dari HTTP.

Testing (dev_test.go)

- a. Fungsi dari dev_test.go adalah untuk melakukan pengujian dari fungsi fungsi dalam projek.
- b. Dev_test.go berisi fungsi yang akan menguji dan memverifikasi fungsi fungsi utama sehingga dapat dipastikan berjalan baik sebelum di *deploy*.

go.mod

- a. File ini adalah komponen integral dari Go *Modules* dan mencatat semua dependensi proyek beserta versi-versi spesifik mereka.
- b. go.mod bertanggung jawab untuk memastikan konsistensi versi dari setiap dependensi yang digunakan dalam proyek.

go.sum

- a. File go.sum berisi *hash checksum* dari setiap dependensi yang digunakan oleh proyek.
- b. Fungsinya adalah untuk memastikan bahwa setiap kali dependensi diunduh, checksum-nya sesuai dengan nilai yang telah disetujui sebelumnya.
- c. Dengan demikian, go.sum berperan penting dalam memastikan konsistensi dan keamanan unduhan dependensi, serta memverifikasi bahwa tidak ada perubahan yang tidak diinginkan pada dependensi yang diunduh

3. Topik

Setelah menguraikan struktur proyek dan mengembangkan *backend* menggunakan GoLang, kita akan mulai pada bagian inti dari projek ini. Dalam sentimen analisis topik merupakan suatu aspek yang krusial. Analisis sentimen topik ini memainkan peran inti dalam membantu memahami dan mengekstrak sentimen dari data teks.

4. Pengambilan Data Youtube

Pengambilan data, atau yang dikenal sebagai "*scraping*," adalah langkah penting dalam proyek ini. Untuk mendapatkan data dari YouTube, pertama-tama identifikasi informasi yang ingin Anda ambil, seperti judul video, deskripsi, komentar, jumlah *like/dislike*. Dalam studi kasus ini kita akan mengambil data komentar. Untuk melakukan *scraping* data kita memerlukan apikey yang bisa didapat dengan membuat projek Google Cloud Platform.

5. Analisa Sentimen

Analisa sentimen merupakan inti dari buku ini, setelah sentimen dianalisis dengan model yang dibuat, data komentar pada *database* akan kita tambah untuk sentimen nya, apakah itu positif, netral atau neutif.

6. Interaksi Dengan *Database*

Dalam implementasi pada projek ini, data topik dan komentar perlu untuk kita simpan. Maka dari itu digunakan *database* untuk menyimpan data yang telah dianalisi sehingga dapat kita lihat Kembali topik yang telah kita buat. Untuk projek ini *database* yang digunakan adalah MongoDB. MongoDB dipilih karena kemampuannya dalam mengelola data yang tidak terstruktur dan fleksibilitasnya dalam menyimpan data dalam format dokumen.

C. UPLOAD BACKEND MENJADI PACKAGE

Setelah semua fungsi selesai, Langkah selanjutnya adalah mengunggah *backend* kita ke pkg.go.dev, sehingga fungsi fungsi yang telah dibuat akan menjadi *package* dan dapat dengan mudah digunakan ketika kita mengintegrasikan dengan google cloud functions.

Langkah pertama adalah *push* semua file tersebut ke github, setelah itu insialisasi versi tag dengan menggunakan
`git tag v0.0.1`

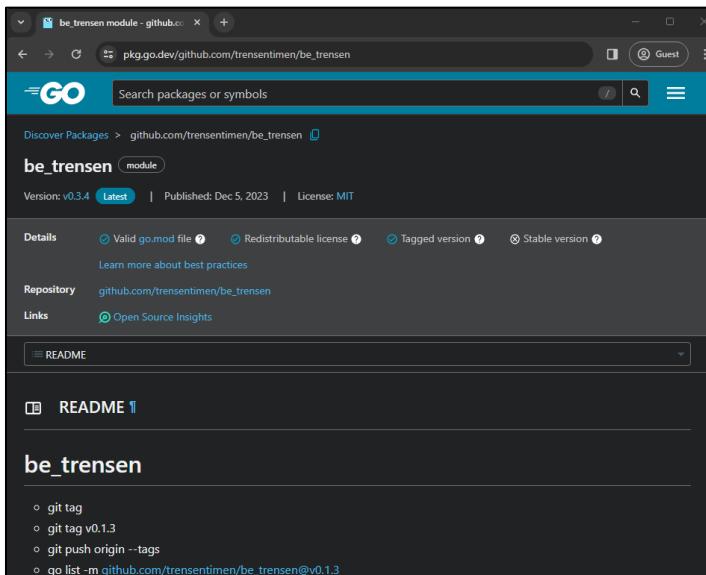
0.0.1 adalah versi yang kita inisialisasi untuk *package* kita. Selanjutnya *push* tag tersebut ke github dengan

```
git push origin -tags
```

Langkah terakhir kita unggah ke pkg.go.dev dengan

```
go list -m github.com/trensentimen/be_trensen@v0.0.1
```

Lalu tunggu beberapa saat dan cek apakah *package* kita terunggah atau tidak dengan mengunjungi link `pkg.go.dev/nama_akun/nama_repo`. Jika berhasil akan tampil seperti ini.



BAB 7

INTEGRASI DENGAN CLOUD FUNCTIONS

A. PENGANTAR CLOUD FUNCTIONS

Cloud functions merupakan layanan komputasi yang tidak menggunakan server, sehingga tetap memungkinkan pengguna untuk membuat serta menjalankan kode program. Disini kami menggunakan Google cloud functions, merupakan suatu layanan yang disediakan oleh Google Cloud Platform dimana secara garis besar memiliki peran dan fungsi yang sama (Perron et al., 2020). Manfaat dari menggunakan cloud functions ini sendiri, sebagai berikut:

1. Skalabilitas, dimana tidak terdapat ketentuan ataupun syarat khusus mengenai infrastruktur yang harus dipenuhi sehingga anda dapat membuat serta menjalankannya.
2. *user friendly*, dimana memiliki pengembang yang sederhana dan reaktif, sehingga memudahkan *user* dalam proses pengerjaannya, dimana *user* tinggal menuliskan kode dan cloud functions yang akan menangani infrastruktur operasionalnya.
3. Ketergantungan terbuka, maksudnya adalah dimana memiliki kemampuan menggunakan kerangka kerja FaaS (function as a service) dari sumber terbuka sehingga bisa dijalankan di berbagai lingkungan sehingga dapat mencegah ketergantungan hanya bisa digunakan pada satu platform tertentu saja.
4. Menghemat pengeluaran, anda tidak perlu mengekuarkan biaya tertentu ketika fungsi yang dibuat sedang aktif ataupun tidak aktif.
5. Juga memiliki prinsip perlindungan akses, seperti:
6. Enkripsi data, dimana data yang disimpan serta diproses dienkripsi baik saat berada dalam penyimpanan maupun ketika akan berpindah melalui jaringan.
7. Pengaturan hak akses pada data, anda dapat mengatur hak akses yang tepat pada Tingkat penyimpanan data misalnya, Google Cloud Storage, Cloud Firestore atau Cloud SQL.

Google Cloud Functions menyediakan kemudahan bagi pengembang dalam mengintegrasikan paket Golang yang telah dibuat ke dalam API. Dengan menggunakan layanan ini, pengguna dapat dengan mudah membuat fungsi-fungsi yang dapat dipanggil langsung dari *frontend*. Hal ini membuka peluang untuk menyederhanakan penggunaan *package* Golang dan model *machine learning* yang memungkinkan integrasi yang lancar antara *backend* dan *frontend*. Proses ini mempercepat pengembangan aplikasi dengan meminimalkan kompleksitas implementasi dan memungkinkan fokus yang lebih besar pada fitur utama. Dengan Google Cloud Functions, pembuatan API dari paket Golang menjadi lebih efisien, mendukung paradigma *serverless*, dan memberikan fleksibilitas yang diperlukan untuk memenuhi kebutuhan pengembangan perangkat lunak saat ini.

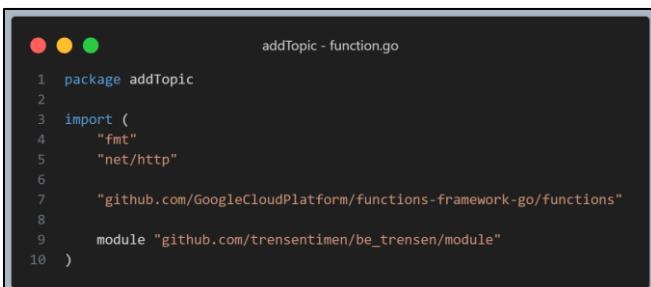
Bagian ini akan menjelaskan bagaimana fungsi *serverless* yang dibuat menggunakan GCF dapat diintegrasikan dengan *backend*. Hal ini memungkinkan pekerjaan tertentu untuk dilakukan secara otomatis dan efisien. Berikut adalah prosedur dan faktor yang harus dipertimbangkan selama proses integrasi.

Code pemograman pada bagian Perancangan API dengan GCF dapat dilihat pada tautan yang terlampir di prakata.

B. IMPLEMENTASI INTEGRASI CLOUD FUNCTIONS PADA BACKEND

kita akan mengintegrasikan package *backend* dengan google cloud functions. Sebagai contoh kita akan membuat function addTopic. Untuk kode dari fungsi lainnya dapat dilihat di repo github yang telah tertera di prakarta.

Pertama kita buat projek golang dengan fungsi **go mod init**. Lalu kita buat function.go, di dalamnya pertama kita import dulu semua package yang diperlukan.



```
addTopic - function.go
1 package addTopic
2
3 import (
4     "fmt"
5     "net/http"
6
7     "github.com/GoogleCloudPlatform/functions-framework-go/functions"
8
9     module "github.com/trensentimen/be_trensen/module"
10 )
```

Kode pada gambar diatas mengimpor paket-paket yang diperlukan, termasuk "fmt" untuk format output, "net/http" untuk berinteraksi dengan

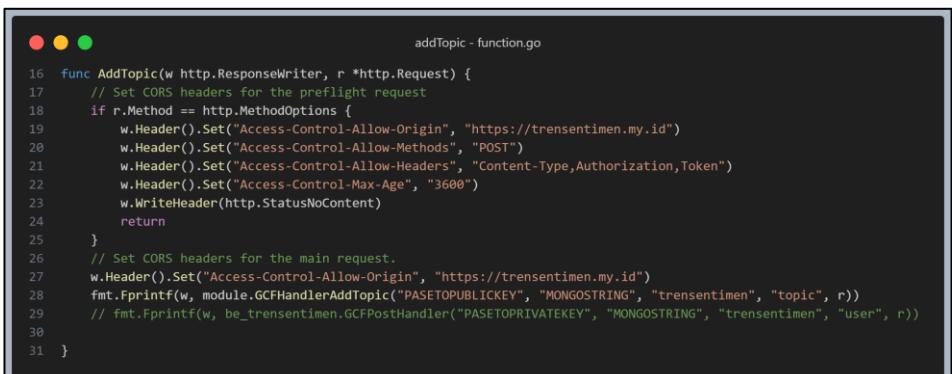
HTTP, "github.com/GoogleCloudPlatform/functions-framework-go/functions" untuk menggunakan Google Cloud Functions, dan paket dengan alias "*module*" dari "github.com/trensentimen/be_trensen/module". Selanjutnya tambahkan fungsi **init()**.



```
addTopic - function.go

12 func init() {
13     functions.HTTP("addTopic", AddTopic)
14 }
```

Fungsi **init()** dijalankan saat paket diimpor dan digunakan untuk mendaftarkan fungsi **AddTopic** sebagai fungsi HTTP dengan nama "addTopic" menggunakan Google Cloud Functions.



```
addTopic - function.go

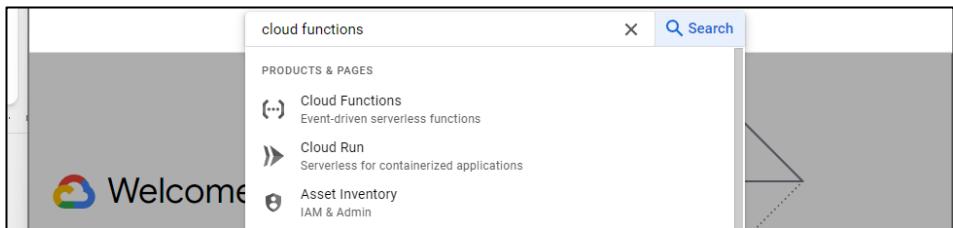
16 func AddTopic(w http.ResponseWriter, r *http.Request) {
17     // Set CORS headers for the preflight request
18     if r.Method == http.MethodOptions {
19         w.Header().Set("Access-Control-Allow-Origin", "https://trensentimen.my.id")
20         w.Header().Set("Access-Control-Allow-Methods", "POST")
21         w.Header().Set("Access-Control-Allow-Headers", "Content-Type,Authorization-Token")
22         w.Header().Set("Access-Control-Max-Age", "3600")
23         w.WriteHeader(http.StatusNoContent)
24         return
25     }
26     // Set CORS headers for the main request.
27     w.Header().Set("Access-Control-Allow-Origin", "https://trensentimen.my.id")
28     fmt.Fprintf(w, module.GCFHandlerAddTopic("PASETOPUBLICKEY", "MONGOSTRING", "trensentimen", "topic", r))
29     // fmt.Fprintf(w, be_trensentimen.GCFPostHandler("PASETOPRIVATEKEY", "MONGOSTRING", "trensentimen", "user", r))
30
31 }
```

Fungsi **AddTopic** menangani permintaan HTTP. Pertama, ia mengatur header CORS sesuai dengan metode permintaan. Selanjutnya, ia memanggil fungsi **GCFHandlerAddTopic** dari modul yang diimpor dengan parameter yang sesuai dan menulis hasilnya ke **http.ResponseWriter**.

Pada Header `Access-Control-Allow-Origin` ganti website dengan domain website yang akan digunakan untuk membangun *frontend*. Header `Access-Control-Allow-Origin` menentukan domain mana yang diizinkan untuk melakukan permintaan ke server. Ini adalah langkah keamanan yang diterapkan oleh browser untuk melindungi sumber daya di server dari permintaan lintas domain yang tidak sah.

Pada kode module.GCFHandlerAddTopic, kode di tersebut memanggil fungsi **GCFHandlerAddTopic** dari paket yang telah dibuat tadi dengan beberapa parameter. Setelah itu, hasilnya dikirimkan sebagai tanggapan HTTP menggunakan fmt.Fprintf.

Selanjutnya buka console.cloud.google.com, jika belum meliki akun buat terlebih dahulu, jika sudah ada klik pencarian dan cari “cloud functions”. Lalu klik “cloud functions”



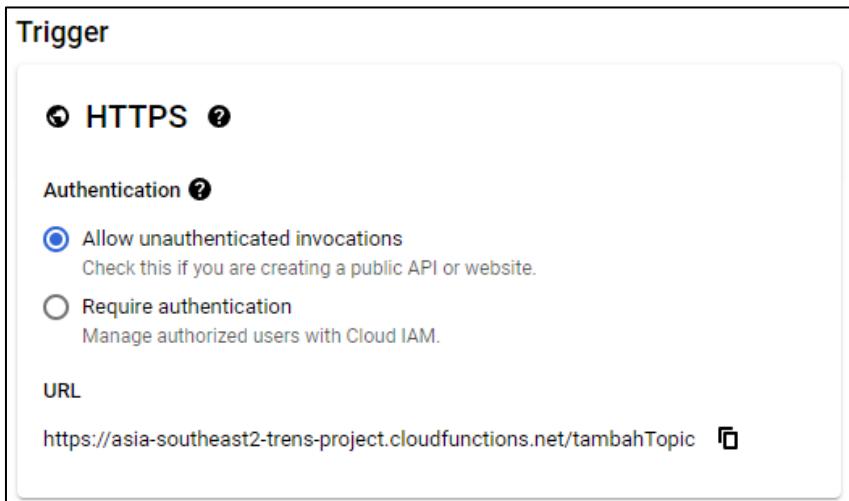
Lalu buat fungsi baru dengan klik “*CREATE FUNCTIONS*” seperti pada gambar dibawah.



Setelah itu pada kolom Environment pilih “2nd gen”, berikan nama fungsi pada Function name dan ubah *Region* menjadi asia-southeast2(Jakarta).

Environment	2nd gen
Function name *	tambahTopic
Region *	asia-southeast2 (Jakarta)

Lalu pada Trigger HTTPS untuk autentikasi pilih *Allow unauthenticated invocation*. Ini berfungsi supaya *cloud function* dapat dipanggil tanpa autentikasi google.



Selanjutnya klik *dropdown Runtime, build, connections and security settings* dan cari Runtime environment variables lalu tambahkan variabel yang diperlukan seperti PUBLICTOKENENV, PRIVATETOKENENV, dan MONGOSTRING. Sesuaikan dengan yang telah dibuat.

The screenshot shows the 'Runtime environment variables' configuration dialog. It contains a single entry: Name 1 * MONGOSTRING and Value 1.

Klik *next*, lalu pilih *Runtime Go 1.21*, dan masukan kode function.go dan go.mod yang telah dibuat sebelumnya. Masukan juga *entry point* sesuai dengan nama yang ada di **init()**.

The screenshot shows the Google Cloud Functions editor interface. At the top, there are dropdown menus for 'Runtime' (set to Go 1.21) and 'Entry point' (set to 'addTopic'). Below these are sections for 'Source code' (set to 'Inline Editor') and file lists ('function.go', 'go.mod'). The main area contains the Go code for the 'addTopic' function:

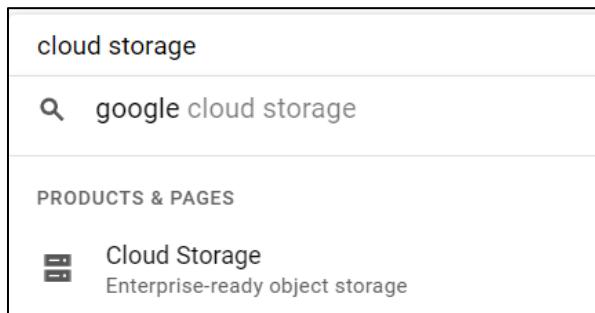
```
Press Alt+F1 for Accessibility Options.
1 package addTopic
2
3 import (
4     "fmt"
5     "net/http"
6
7     "github.com/GoogleCloudPlatform/functions-framework-go/functions"
8
9     module "github.com/trensentimen/be_trensen/module"
10 )
11
12 func init() {
13     functions.HTTP("addTopic", AddTopic)
14 }
15
16 func AddTopic(w http.ResponseWriter, r *http.Request) {
17     // Set CORS headers for the preflight request
18     if r.Method == http.MethodOptions {
19         w.Header().Set("Access-Control-Allow-Origin", "https://trensentimen.my.id")
20         w.Header().Set("Access-Control-Allow-Methods", "POST")
21         w.Header().Set("Access-Control-Allow-Headers", "Content-Type,Authorization-Token")
22         w.Header().Set("Access-Control-Max-Age", "3600")
23         w.WriteHeader(http.StatusOK)
24     }
25     // Set CORS headers for the main request.
26     w.Header().Set("Access-Control-Allow-Origin", "https://trensentimen.my.id")
27     fmt.Println("w, module.GCFHeaderAddTopic(\"PASETOPUBLIKER\", \"NOHOSTNAME\", \"trensentimen\", \"topic\", r)")
28     // http.Println(w, module.GCFPostHandler("PASETOPRIVATEKEY", "NOHOSTNAME", "trensentimen", "user", r))
29
30 }
31 }
```

Kemudian klik *Deploy*. Tunggu hingga selesai dan proses *deploy* sukses. Jika berhasil sukses maka akan ada tanda ceklis hijau pada halaman *console* utama.



C. IMPLEMENTASI INTEGRASI CLOUD FUNCTIONS PADA MODEL ML

Selanjutnya kita akan membuat cloud functions untuk integrasi model. Pertama upload dulu model serta vectorisasi model ke dalam google storage. Pertama kita cari cloud storage di kolom pencarian, lalu klik cloud storage.



Pertama kita buat bucket untuk menyimpan model dan vektorisasi. Klik *Create* untuk membuat bucket



Masukan nama *bucket* dan klik *continue*.

A screenshot of the "Name your bucket" creation step. It shows a text input field containing "trensentimen-bucket-1". Below the input field is a tip: "Tip: Don't include any sensitive information". There is also a "LABELS (OPTIONAL)" section which is currently collapsed. At the bottom is a "CONTINUE" button.

Lokasi pilih *asia (multiple regions in Asia)*.

A screenshot of the "Choose where to store your data" step. A checkmark is next to the title "Choose where to store your data". Below the title, a descriptive text states: "This choice defines the geographic placement of your data and affects cost, performance, and availability. Cannot be changed later." followed by a "Learn more" link. Under the heading "Location type", there are three options: "Multi-region" (selected, indicated by a blue circle), "Dual-region", and "Region". Each option has a brief description below it. A dropdown menu is open, showing "asia (multiple regions in Asia)" as the selected value. At the bottom is a "CONTINUE" button.

Pilih "Set a default class" lalu pilih "standard", Klik *continue*.

- Choose a storage class for your data

A storage class sets costs for storage, retrieval, and operations, with minimal differences in uptime. Choose if you want objects to be managed automatically or specify a default storage class based on how long you plan to store your data and your workload or use case. [Learn more](#)

Autoclass

Automatically transitions each object to Standard or Nearline class based on object-level activity, to optimize for cost and latency. Recommended if usage frequency may be unpredictable. Can be changed to a default class at any time. [Pricing details](#)

Set a default class

Applies to all objects in your bucket unless you manually modify the class per object or set object lifecycle rules. Best when your usage is highly predictable. Can't be changed to Autoclass once the bucket is created.

Standard

Best for short-term storage and frequently accessed data

Nearline

Best for backups and data accessed less than once a month

Coldline

Best for disaster recovery and data accessed less than once a quarter

Archive

Best for long-term digital preservation of data accessed less than once a year

[CONTINUE](#)

Uncheck “Enforce public access prevention on this bucket”.

- Choose how to control access to objects

Prevent public access

Restrict data from being publicly accessible via the internet. Will prevent this bucket from being used for web hosting. [Learn more](#)

Enforce public access prevention on this bucket

Access control

Uniform

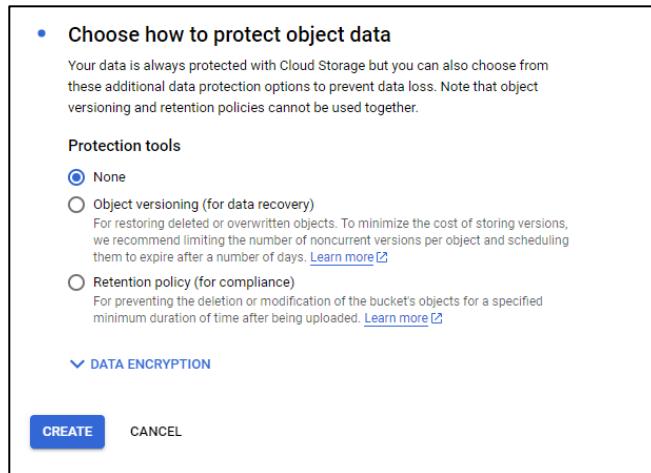
Ensure uniform access to all objects in the bucket by using only bucket-level permissions (IAM). This option becomes permanent after 90 days. [Learn more](#)

Fine-grained

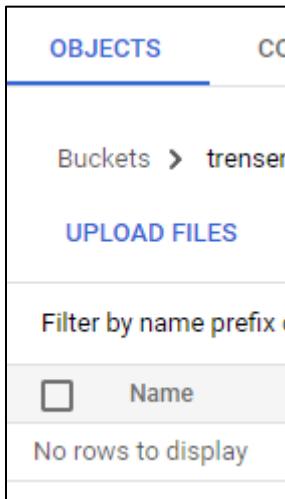
Specify access to individual objects by using object-level permissions (ACLs) in addition to your bucket-level permissions (IAM). [Learn more](#)

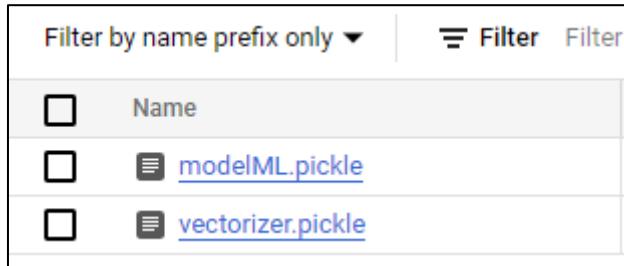
[CONTINUE](#)

Pilih none dan klik *create*



Upload file model dan file vektorisasi.





Setelah itu kita buat cloud functions, buat cloud functions seperti sebelumnya, ganti runtime dengan python

PREVIOUS DEPLOY CANCEL

Berikut adalah penjelasan dari kode cloud function pada gambar diatas.

 analizeSentimen - function.py

```
1 import functions_framework  
2 import pickle  
3 from google.cloud import storage  
4 import sklearn
```

Pada gambar diatas merupakan Kode untuk mengimpor beberapa pustaka yang diperlukan, termasuk Functions Framework untuk Python, modul pickle untuk serialisasi objek, google.cloud.storage untuk berinteraksi dengan Google Cloud Storage, dan modul sklearn yang digunakan untuk *machine learning*.

 analizeSentimen - function.py

```
6 @functions_framework.http  
7 def model(request):
```

Fungsi ini di-decorate dengan @functions_framework.http untuk menandakan bahwa ini adalah fungsi HTTP yang akan ditangani oleh Google Cloud Functions.



analizeSentimen - function.py

```
10 if request.method == 'OPTIONS':
11     headers = {
12         "Access-Control-Allow-Origin": "https://trensentimen.my.id",
13         "Access-Control-Allow-Methods": "POST",
14         "Access-Control-Allow-Headers": "Content-Type",
15         "Access-Control-Max-Age": "3600",
16     }
17     # Handle preflight requests
18     return ('', 204, headers)
19 headers = {"Access-Control-Allow-Origin": "https://trensentimen.my.id"}
```

Kode ini menangani permintaan preflight (OPTIONS) dengan mengatur header CORS untuk memungkinkan permintaan dari domain tertentu.



analizeSentimen - function.py

```
44 else:
45     response_data = {
46         "message": False,
47         "data": []
48 }
```

Kode ini membentuk respons JSON dengan status *false* pada saat terjadi kesalahan.



analizeSentimen - function.py

```
8 request_json = request.get_json(silent=True)
9 request_args = request.args
```

Kode ini mengambil data JSON dari permintaan dan argumen permintaan. `get_json(silent=True)` digunakan untuk mengambil payload JSON dari permintaan.

```
● ● ● analizeSentimen - function.py  
23 storage_client = storage.Client()  
24 bucket = storage_client.get_bucket('trensentimen_bucket-1')  
25 blob_classifier = bucket.blob('modelNB6.pickle')
```

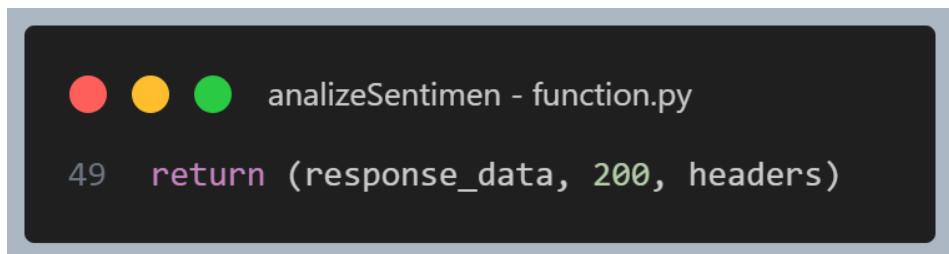
Kode pada gambar diatas berfungsi untuk membuat koneksi terhadap google storage dan mengunduh file model dan vektorisasi dari google cloud storage.

```
● ● ● analizeSentimen - function.py  
36 # Konversi teks menjadi vektor menggunakan CountVectorizer  
37 new_text_vectorized = serverless_vectorizer.transform(sentiments)  
38 # Make predictions using the loaded classifier model  
39 dataSentiments = serverless_classifier.predict(new_text_vectorized)
```

Kode ini menggunakan model dan vectorizer yang telah didownload untuk melakukan prediksi sentimen terhadap teks yang diberikan

```
● ● ● analizeSentimen - function.py  
40 response_data = {  
41     "message": True,  
42     "data": dataSentiments.tolist()  
43 }
```

Kode ini membentuk respons JSON dengan status true dan hasil prediksi sentimen.



The screenshot shows a code editor window with a dark theme. At the top, there are three colored circular icons: red, yellow, and green. To the right of these icons, the file name "analizeSentimen - function.py" is displayed. Below the icons, the code is shown in a monospaced font:

```
49     return (response_data, 200, headers)
```

Gamabar diatas merupakan kode untuk mengembalikan data respon sebagai respons HTTP dengan kode status 200 dan header CORS yang sesuai.

Kode ini sepertinya dirancang untuk memproses permintaan HTTP yang berisi data sentimen, melakukan prediksi menggunakan model yang telah *di-train* sebelumnya, dan mengembalikan hasil prediksi sebagai respons JSON. Respons juga mencakup header CORS untuk memungkinkan permintaan dari domain tertentu.

BAB 8

MEMBANGUN *FRONTEND*

A. PENDAHULUAN

Yang dimaksud dari membangun *frontend* merupakan proses membuat rancangan, proses mengembangkan, serta proses mengaplikasikan ui/ux dari suatu aplikasi yang sedang dibangun. Didalam ui/ux itu terdapat rancangan desain yang telah dibuat, serta melibatkan gabungan dari beberapa elemen seperti tata letak, garis, navigasi, grafis, dll (Oktiriani et al., 2022).

Singkatnya *frontend* itu merupakan wajah dari aplikasi, yang dimana menjadi kontak antara *user* dengan sistem. Dalam *frontend* itu sendiri ada beberapa aspek penting yang perlu di perhatikan ketika ingin membangun sebuah *frontend* seperti:

1. Tata letak atau layout, karena susunan elemen visual dan navigasi yang memungkinkan pengguna menavigasi aplikasi atau situs web dengan mudah.
2. Desain responsive, dimana antarmuka yang ditampilkan harus disesuaikan dengan berbagai perangkat, dimana ukuran pada tiap perangkat itu berbeda-beda. Sehingga jika itu sudah disesuaikan akan memberikan kenyamanan kepada *user* pada saat menggunakan aplikasi yang dibangun.
3. Kecepatan dan kinerja, antarmuka yang dibangun harus semaksimal mungkin bisa berfungsi secara efektif, efisien serta responsif tanpa delay yang dapat menyebabkan penurunan kualitas penilaian aplikasi.
4. Interaktivitas, yang dimaksud adalah seperti memberikan animasi, validasi, serta berbagai macam tindakan yang bisa dilakukan oleh *user*, sehingga dapat memberikan respon dan interaksi dalam waktu yang nyata bagi pengguna.

Dalam proses pengembangannya sendiri *frontend* dapat melibatkan berbagai bahasa pemrograman, framework, serta teknologi. Yang umum digunakan itu seperti: HTML, CSS, JavaScript, Framework *Frontend*.

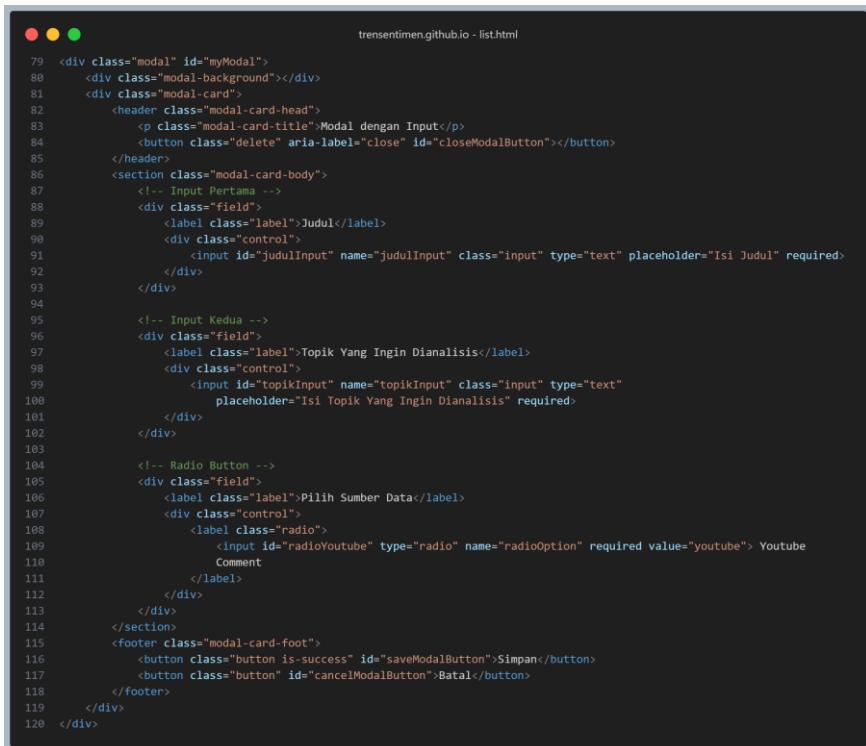
Sedangkan dalam pengembangannya melibatkan beberapa tahapan seperti: perancangan, desain, pembuatan prototype, pengkodean, implementasi, pengujian serta optimalisasi.

B. IMPLEMENTASI

1. Fetch API

Setelah membuat API menggunakan Google Cloud Functions, sekarang kita akan mencoba menggunakan memanggil api tersebut di *frontend* dengan menggunakan fungsi javascript yaitu **fetch**. Disini kita akan mencoba untuk memanggil api untuk menambahkan topik.

Pertama kita buat file html, lalu buat *form* untuk menambahkan data. Disini kita menggunakan modal untuk membuat *form* tambah topik.



```
trentsentimen.github.io - list.html
79 <div class="modal" id="myModal">
80   <div class="modal-background"></div>
81   <div class="modal-card">
82     <header class="modal-card-head">
83       <p class="modal-card-title">Modal dengan Input</p>
84       <button class="delete" aria-label="Close" id="closeModalButton"></button>
85     </header>
86     <section class="modal-card-body">
87       <!-- Input Pertama -->
88       <div class="field">
89         <label class="label">Judul</label>
90         <div class="control">
91           <input id="judulInput" name="judulInput" class="input" type="text" placeholder="Isi Judul" required>
92         </div>
93       </div>
94
95       <!-- Input Kedua -->
96       <div class="field">
97         <label class="label">Topik Yang Ingin Dianalisis</label>
98         <div class="control">
99           <input id="topikInput" name="topikInput" class="input" type="text" placeholder="Isi Topik Yang Ingin Dianalisis" required>
100        </div>
101      </div>
102
103      <!-- Radio Button -->
104      <div class="field">
105        <label class="label">Pilih Sumber Data</label>
106        <div class="control">
107          <label class="radio">
108            <input id="radioYoutube" type="radio" name="radioOption" required value="youtube"> Youtube
109            Comment
110          </label>
111        </div>
112      </div>
113    </div>
114  </section>
115  <footer class="modal-card-foot">
116    <button class="button is-success" id="saveModalButton">Simpan</button>
117    <button class="button" id="cancelModalButton">Batal</button>
118  </footer>
119 </div>
120 </div>
```

Pada gambar diatas adalah untuk membuat *form* modal,sekarang kita buat button untuk memunculkan *form* tersebut.

```
trentsentimen.github.io - list.html  
161 <button class="button is-warning is-rounded fixed-button is-large" id="showModalButton">  
162     <ion-icon name="add-outline"></ion-icon>  
163 </button>
```

Lalu pada gambar diatas adalah untuk membuat button untuk membuat tombol untuk menampilkan *form*.

```
trentsentimen.github.io - list.html  
181 <script type="module">  
182     import postAddTopic from "js/addTopic.js";  
183     const showModalButton = document.getElementById('showModalButton');  
184     const closeModalButton = document.getElementById('closeModalButton');  
185     const cancelModalButton = document.getElementById('cancelModalButton');  
186     const simpanModalButton = document.getElementById('saveModalButton');  
187     const myModal = document.getElementById('myModal');  
188  
189     showModalButton.addEventListener('click', () => {  
190         myModal.classList.add('is-active');  
191     });  
192  
193     closeModalButton.addEventListener('click', () => {  
194         myModal.classList.remove('is-active');  
195     });  
196  
197     cancelModalButton.addEventListener('click', () => {  
198         myModal.classList.remove('is-active');  
199     });  
200  
201     simpanModalButton.addEventListener('click', () => {  
202         postAddTopic();  
203     });  
204 </script>
```

Lalu pada gambar diatas merupakan fungsi javascript untuk memunculkan dan menghilangkan modal *form*. Diatas terlihat kita mengimport file js bernama addFile.js, file tersebut untuk menyimpan.

```
trentsentimen.github.io - list.html  
10 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bulma@0.9.4/css/bulma.min.css" />
```

Karena kita menggunakan *framework* css dalam pembuatan modal, kita import dulu cdn dari bulma di header.

```
trentsentimen.github.io - list.html
178 <script type="module" src="https://unpkg.com/ionicons@7.1.0/dist/ionicons/ionicons.esm.js"></script>
179 <script nomodule src="https://unpkg.com/ionicons@7.1.0/dist/ionicons/ionicons.js"></script>
```

Lalu kita import cdn untuk icon.

```
trentsentimen.github.io - addTopic.js
1 import { postWithBearer } from "./utilities/api.js";
2 import getCookie from "./getCookie.js";
3 import { getValue } from "./utilities/element.js"
```

Sekarang kita buat file js untuk memanggil API untuk menambahkan topic. Pertama kita import dulu file yang diperlukan. postWithBearer adalah fungsi untuk Fetch API, getCookie untuk mendapatkan Cookie, dan Getvalue adalah untuk mendapatkan nilai dari *form* di file html. Untuk detail fungsi tersebut dapat dilihat pada github yang tertera di prakarta.

```
trentsentimen.github.io - addTopic.js
4
5 const PostAddTopic = () => {}
```

Lalu kita buat fungsi dengan nama PostAddTopic.

```
trentsentimen.github.io - addTopic.js
6 const radioButtons = document.querySelectorAll('input[name="radioOption"]');
7 let selectedValue;
8
9 radioButtons.forEach radioButton => {
10   if (radioButton.checked) {
11     selectedValue = radioButton.value;
12   }
13 });
14
15 //validasi form
16 if (getValue("judulInput") === "" || getValue("judulInput") === null) {
17   alert("Topik tidak boleh kosong");
18   return;
19 } else if (getValue("topikInput") === "" || getValue("judulInput") === null) {
20   alert("Topik tidak boleh kosong");
21   return;
22 } else if (selectedValue === "" || selectedValue === null) {
23   alert("Topik tidak boleh kosong");
24   return;
25 }
```

Lalu dalam fungsi tersebut tambahkan kode untuk mendapatkan nilai dari *form* di file html seperti pada gambar diatas.

```
● ○ ● trentsentimen.github.io - addTopic.js
27 const target_url = "https://asia-southeast2-trens-project.cloudfunctions.net/addTopic";
28 const datainjson = {
29   topicname: getValue("judulInput"),
30   source: {
31     source: selectedValue,
32     value: getValue("topikInput")
33   }
34 };
35 const token = getCookie("token")
36 if (token) {
37   postWithBearer(target_url, token, datainjson, responseData)
38   alert("Berhasil menambahkan topik")
39 } else {
40   console.log("token tidak ada " + result.message);
41   alert("sesi anda sudah habis, silahkan logout dan login ulang")
42 }
```

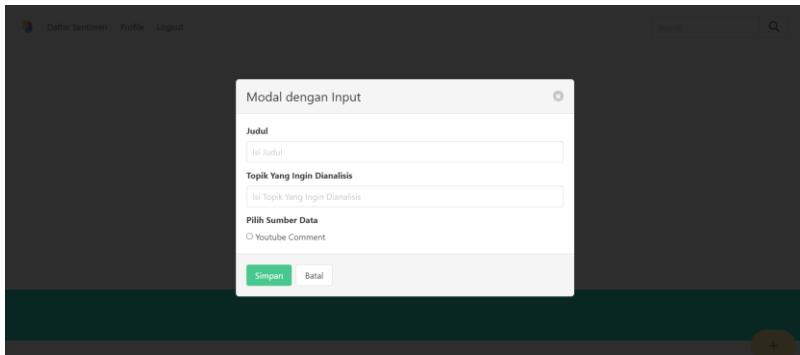
Lalu dibawahnya tambahkan kode untuk memanggil API addTopic.

```
● ○ ● trentsentimen.github.io - addTopic.js
45 const responseData = (result) => {
46   // console.log(result);
47   if (result.status === true) {
48     window.location.href = "sentimen.html";
49   } else {
50     console.log(result.message);
51     alert(`sesi anda sudah habis, silahkan logout dan login ulang`);
52   }
53 };
```

Lalu dibawah fungsi PostAddTopic buat fungsi untuk memvalidasi return dari API.



Berikut adalah tampilan file html yang telah dibuat.



Ketika tombol tambah ditekan maka akan muncul modal yang berisi form, lalu jika tombol simpan ditekan maka akan memanggil fungsi postAddTopic.

A detailed view of the "Modal dengan Input" form. The "Judul" field contains "Test Get Comment Youtube". The "Topik Yang Ingin Dianalisis" field contains "-XMZEweNSfE". Under "Pilih Sumber Data", the radio button for "Youtube Comment" is selected. At the bottom are the "Simpan" and "Batal" buttons.

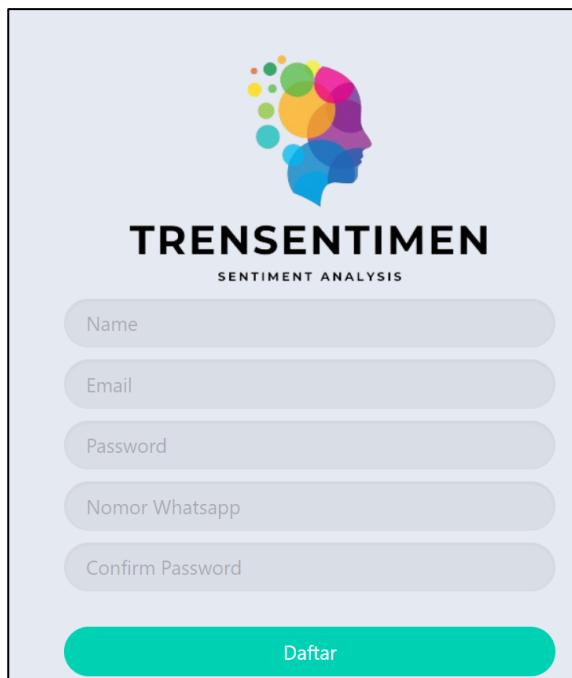
Sekarang kita tes dengan mencoba untuk menambahkan data topik.

```
_id: ObjectId('65758cfcd4aa4625ff3f333')
topicname: "Test Get Comment Youtube"
source: Object
  source: "youtube"
  value: "-XMZEweNSfE"
  date_range: ""
status: "drafting"
```

Pada gambar diatas merupakan data yang tersimpan pada *database* mongo.

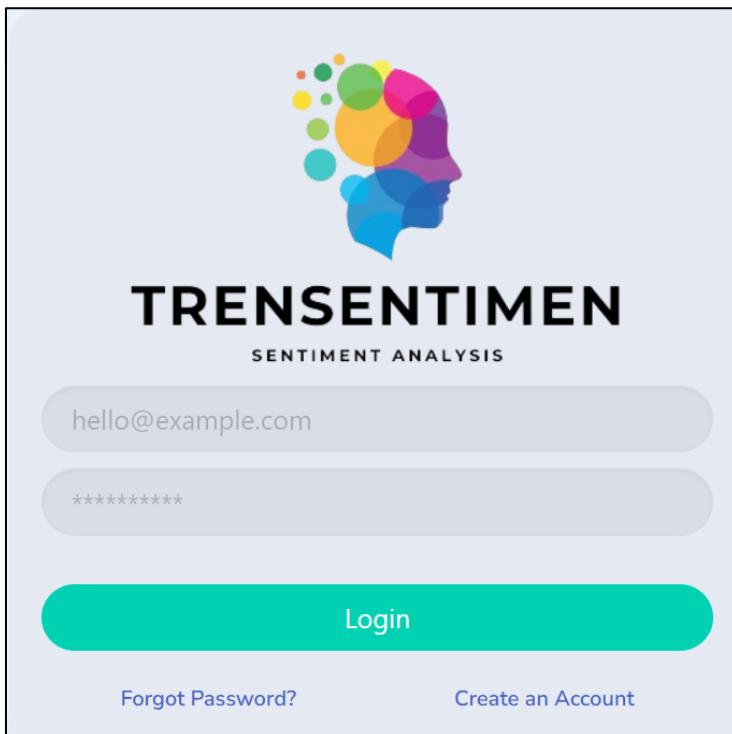
2. Halaman Register

Halaman register adalah halaman untuk menambah *user* dengan inputan nama, email, nomor telepon dan *password*. Pada halaman ini juga memanggil API untuk detail register kode dapat dilihat di github yang tertera di prakata.



3. Halaman Login

Halaman login adalah halaman untuk masuk ke dalam website dengan inputan email dan *password*. Pada halaman ini juga memanggil API untuk detail kode login dapat dilihat di github yang tertera di prakata.



4. Halaman Daftar Topik

Halaman daftar topik menampilkan topik apa saja yang telah *user* inputkan dan terdapat tombol untuk menambah topik baru seperti pada bagian pembahasan fetch API pada point 1 diatas. untuk detail kode dapat dilihat di github yang tertera di prakata.

test youtube youtube	test youtube2 youtube	pilpress youtube
"Sesuatu Di Jogja" - Adhitia Sofyan (official audio) youtube	Wawancara Jokowi youtube	Test Get Comment Youtube youtube

5. Add Topic

AddTopic adalah modal yang berisi *form* untuk menambah topik, untuk detail dapat dilihat pada point 1 diatas atau yang lebih lengkap dapat dilihat pada github yang tertera di prakata.

The modal window has a title bar 'Modal dengan Input' and a close button. It contains three input fields: 'Judul' (Title) with placeholder 'Isi Judul', 'Topik Yang Ingin Dianalisis' (Topic to be analyzed) with placeholder 'Isi Topik Yang Ingin Dianalisis', and 'Pilih Sumber Data' (Select Data Source) with two radio buttons: 'X (Twitter)' and 'Youtube Comment'. At the bottom are two buttons: a green 'Simpan' (Save) button and a white 'Batal' (Cancel) button.

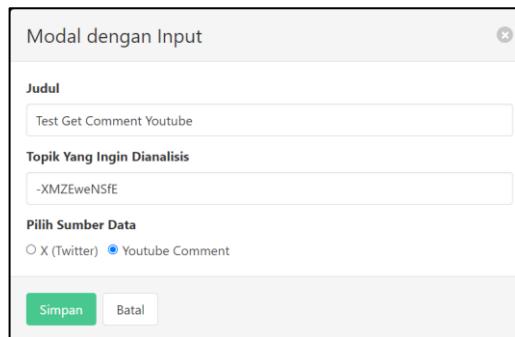
6. Halaman Detail Topik

Halaman ini merupakan detail topik yang telah kita masukan, dihalaman ini juga ada fungsi untuk mengubah topik dan menghapus topik. Lalu terdapat juga fungsi untuk mendapatkan data comment youtube dan fungsi untuk menganalisa sentiment pada setiap komen. Dihalaman ini juga ada grafik yang menampilkan jumlah presentasi sentiment ketika telah dilakukan analisis. Untuk detail kode dapat dilihat pada github yang tertera di prakata.

The detail page displays the following information: 'Judul: Test Get Comment Youtube', 'Topic: -XMZEweNSfE', and 'Sumber Data: youtube'. At the bottom are two buttons: a red 'Delete' button and a blue 'Scrap Data' button.

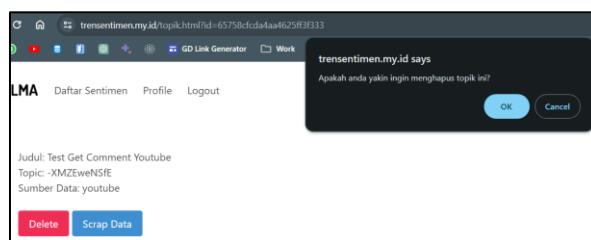
7. Update Topic

Update topic merupakan fungsi untuk melakukan perubahan pada topik yang telah diinput. Untuk detailnya dapat dilihat pada github yang tertera di prakata.



8. Delete Topic

Merupakan fungsi untuk menghapus data topik. Untuk detailnya dapat dilihat pada github yang tertera di prakata.



9. Get Data

Setelah menambahkan topik, lalu tambahkan tombol untuk menadapatkan data komentar youtube dengan API untuk Scrap data. Untuk detailnya dapat dilihat pada github yang tertera di prakata.

No	Text	Sentimen
1	Saya senang Bapak selalu bilang pakai 'kami' kalau mau cerita relasi jejak. Karena waktu itu pas mengerjakan, gak sendirian. Dibantu sama timnya (Walikota.Bupati, Kadis dan jajaran)	Belum di analisa
2	Om Refly, Om Ganjar waktunya kuliah di UGM aktivitinya apa aja? Undang dong ke podcatnya...	Belum di analisa
3	baru kali ini saya lihat Refly Harun dan Rocky Gedung berbinar-binar sejak adanya harapan sosok presiden masa depan seperti Anies Baswedan. Mereka satu frekuensi..	Belum di analisa
4	Senomoga rakyat Indonesia semakin cerdas pengetahuan politik nya, sehingga tdk bisa lagi ditakut-takuti dan di beli dgn uang suara nya di pilpres 2024 ini, Anies Baswedan dg Muhammadi pilihan kita, amin yarobbul alaamin, trim's	Belum di analisa
5	In Syaa Allah Pa Anies RI 1, boleh tuh bang RIH menhukum nya, bang RG penasehatnya.	Belum di analisa

10. Analisis Data

Setelah mendapatkan data komentar youtube maka kita analisis dengan model yang sudah kita unggah di Api Cloud Function. Pada proses nya setelah analisis data respon nya adalah memanggil lagi API untuk mengubah data sentiment di *database*. Untuk detailnya dapat dilihat pada github yang tertera di prakata.

No	Text	Sentimen
1	Saya senang Bapak selalu bilang pakai 'kami' kalau mau cerita rekam jejak. Karena waktu itu pas mengerjakan, gak sendirian. Dibantu sama timnya (Walikota,Bupati, Kadis dan jajaran)	negative
2	Om Refly, Om Ganjar waktu kuliah di UGM aktivitasnya apa aja? Undang dong ke podcastnya...	negative
3	baru kali ini saya lihat Refly Harun dan Rocky Gedung berbinar-binar sejak adanya harapan sosok presiden masa depan seperti Anies Baswedan. Mereka satu frekuensi...	positive
4	Semoga rakyat Indonesia semakin cerdas pengetahuan politik nya, sehingga tdk bisa lagi ditakut-takuti dan di beli dgn uang suara nya di pilpres 2024 ini; Anies Baswedan dg Muhamimin pilihan kita, aamin yarobbul alamin, trim's	negative
5	In Syaa Allah Pa Anies RI 1, boleh tuh bang RH menhukum nya, bang RG penasehatnya.	negative

BAB 9

MENGUNGGAH JS MENCADI CDN

A. PENDAHULUAN

Salah satu metode yang umum digunakan guna meningkatkan kinerja serta aksesibilitas JavaScript adalah dengan menggunakan jaringan pengiriman konten (CDN). Penggunaan CDN untuk mendistribusikan JavaScript merupakan praktik yang melibatkan *uploading* file JavaScript ke infrastruktur CDN yang tersebar diseluruh dunia. Ini mungkin file yang sama diakses oleh berbagai pengguna dari berbagai lokasi serta waktu *response* sesingkat mungkin (Nakhaei et al., 2020). Mengunggah JavaScript ke CDN menawarkan beberapa keuntungan yang signifikan, seperti:

1. Percepatan Akses, dimana *user* bisa mengakses file JavaScript dari lokasi server terdekat, sehingga dapat mengurangi waktu pemuatan halaman.
2. Ketersediaan Global, dimana infrastruktur CDN yang tersedia itu tersebar diberbagai lokasi geografis, serta dapat memastikan file JavaScript yang ada dalam waktu yang singkat di berbagai lokasi yang ada di dunia.
3. Lonjakan lalu lintas, CDN bisa menaggulangi lonjakan lalu lintas yang terjadi dengan cara mengalokasikan sumber daya secara dinamis.
4. Manajemen dan pembaruan yang mudah, CDN dapat mengelola serta memperbarui versi Javascript dapat dengan cepat dilakukan.

B. IMPLEMENTASI

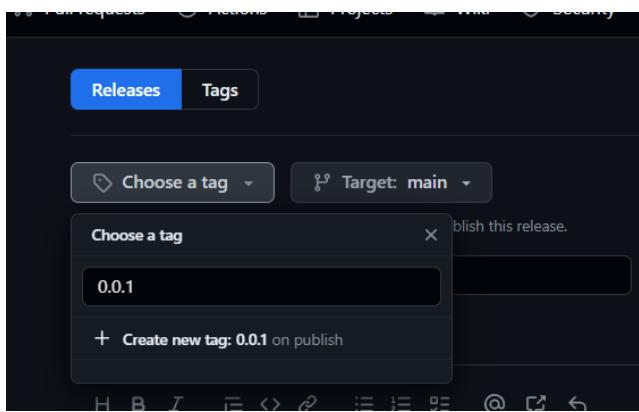
Untuk memindahkan semua file js yang ada pada *frontend*, pertama kita buat repo khusus pada github untuk menyimpan semua file js. Lalu masukkan folder js yang ada dalam *frontend*.

```
> temp
> utilities
  ↗ addTopic.js
  ↗ analyze.js
  ↗ cekCookie.js
  ↗ dashboard.js
  ↗ deleteCookie.js
  ↗ deleteData.js
  ↗ getCookie.js
  ↗ getTopic.js
  ↗ index.html
  ↗ listTopic.js
  ↗ login.js
  ↗ README.md
  ↗ resetPassword.js
  ↗ scrap.js
  ↗ sendOTP.js
  ↗ setCookie.js
  ↗ signup.js
  ↗ test.js
  ↗ updateTopic.js
  ↗ verifyOTP.js
```

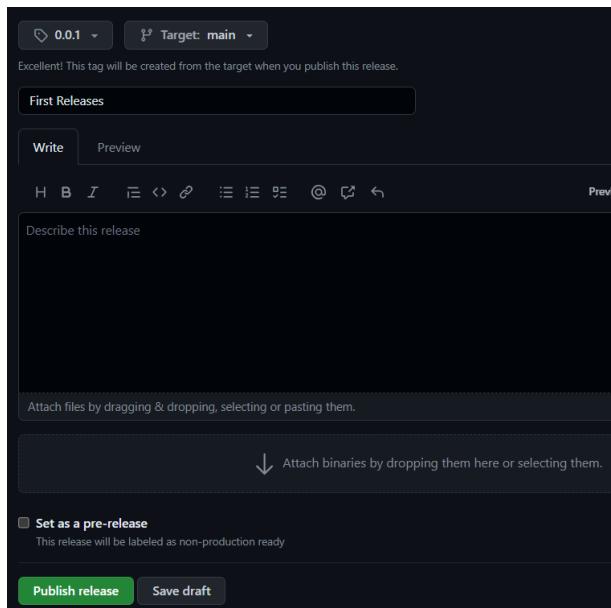
Setelah itu push, lalu buat realise tag.

The screenshot shows a GitHub repository named "js_trensentimen". The repository has 1 branch and 0 tags. The commit history shows 2 commits from "daniferdinandall" with the message "update". The commits are listed with their file names, update times, and commit times. On the right side of the page, there is an "About" section with a "No description, website, or topics provided" message, and a "Releases" section with a "Create a new release" button highlighted by a red circle.

Klik Choose a tag, lalu buat versi tag dan klik **Create new tag: {versi tag}** dibawahnya.



Isi judul lalu *publish* dengan menekan tombol **Publish release** dibawah.



Langkah selanjutnya buka website js deliver dengan tambahan Alamat repo js yang telah dibuat dengan format seperti dibawah ini.

<https://www.jsdelivr.com/package/gh/usernamegithub/repojs>

Contoh:

https://www.jsdelivr.com/package/gh/trentsentimen/js_trentsentimen

The screenshot shows the npm package page for 'trentsentimen/js_trentsentimen'. At the top, there's a search bar with 'search all of npm' and a logo. Below the search bar, the package name 'trentsentimen/js_trentsentimen' is displayed, along with its version '0.0.1'. There are tabs for 'Readme', 'Files' (which is selected), and 'Statistics'. A 'Browse CDN' link is also present. The main content area shows the file structure under 'trentsentimen/js_trentsentimen@0.0.1':

- /gh/trentsentimen/js_trentsentimen@0.0.1/temp
- /gh/trentsentimen/js_trentsentimen@0.0.1/utilities
- /gh/trentsentimen/js_trentsentimen@0.0.1/addTopic.js
- /gh/trentsentimen/js_trentsentimen@0.0.1/analyze.js
- /gh/trentsentimen/js_trentsentimen@0.0.1/cekCookie.js

Each file entry has a download icon and a toggle switch.

Lalu akan tampil halaman daftar CDN dari semua file js yang berada pada repo js. Langkah terakhir adalah mengganti semua import file js di *frontend* dengan import CDN.

The screenshot shows a browser console with three colored status indicators (red, yellow, green) and the URL 'trentsentimen.github.io - list.html'. The console displays the following code:

```
182 import postAddTopic from "js/addTopic.js";
```

Below this, another line of code is shown:

```
182 import postAddTopic from "https://cdn.jsdelivr.net/gh/trentsentimen/js_trentsentimen@0.0.1/addTopic.js";
```

BAB 10

KESIMPULAN

Buku ini telah membahas secara mendalam Langkah Langkah untuk membangun sistem analisis sentimen dengan pendekatan teknologi termasuk *webservice*, *cloud computing*, dan *machine learning* dengan algoritma Naïve Bayes Classifier dalam Python. *Frontend* menggunakan HTML, Bulma CSS, dan Javascript Es6 dengan peningkatan kebersihan kode melalui CDN seperti Jsdelivr. *Backend* menggunakan Golang untuk mengelola *database* dan memanggil model, diintegrasikan melalui *webservice* Google Cloud Function sebagai API untuk mempermudah pemanggilan dari *frontend*. Dengan adanya buku ini diharapkan pembaca dapat memahai konsep mengenai sistem *webservice* dan pemodelan *machine learning*.

A. KESIMPULAN

Dalam penutup, buku tutorial ini telah memberikan pemahaman mendalam tentang prinsip-prinsip dasar dalam membangun model *machine learning*, menjelaskan langkah-langkah yang diperlukan untuk membentuk dasar pengetahuan dalam domain ini. Melalui implementasi algoritma *machine learning*, khususnya Naïve Bayes Classifier dalam bahasa pemrograman Python, pembaca diharapkan dapat memahami secara praktis bagaimana membangun dan melatih model untuk analisis sentimen.

Selanjutnya, tutorial ini membahas proses membangun sebuah *webservice* menggunakan bahasa pemrograman Golang, menyoroti pentingnya efisiensi dan kehandalan dalam mengelola *backend*. Integrasi dengan Google Cloud Functions menambah dimensi kemudahan dan fleksibilitas dalam menyediakan layanan *backend*, memastikan responsivitas sistem.

Langkah terakhir tutorial membahas pemanfaatan API yang dibangun pada *frontend*. Kesimpulannya, pembaca diberikan gambaran praktis tentang cara menghubungkan antara *frontend* dan *backend* menggunakan API, yang merupakan komponen kunci dalam membangun sistem yang terintegrasi dan responsif.

B. SARAN

1. Pengembangan *Machine learning*:
 - Lanjutkan eksplorasi model *machine learning* lainnya selain Naïve Bayes Classifier, seperti model *deep learning* untuk analisis sentimen yang lebih kompleks.
 - Perluas cakupan dataset untuk meningkatkan keakuratan dan generalisasi model.
 - Pertimbangkan untuk menyelidiki teknik fine-tuning dan optimasi model untuk hasil yang lebih baik.
2. Pengembangan *Webservice* menggunakan Golang:
 - Perluas fungsionalitas *webservice* dengan menambahkan fitur-fitur baru yang relevan.
 - Pertimbangkan penggunaan framework Golang yang tersedia untuk mempercepat pengembangan dan meningkatkan skalabilitas.
 - Integrasikan manajemen kesalahan dan pelaporan untuk memudahkan pemecahan masalah.
3. Mengintegrasikan *Backend* dengan Google Cloud Functions:
 - Eksplorasi berbagai layanan Google Cloud untuk memperkaya kemampuan *backend*.
 - Pertimbangkan penggunaan alat pemantauan dan analisis kinerja untuk memahami dan meningkatkan respons sistem.
 - Implementasikan otentifikasi dan keamanan tambahan untuk melindungi layanan cloud dan data.
4. Pengembangan Antarmuka
 - Tingkatkan antarmuka pengguna dengan desain yang lebih responsif dan interaktif.
 - Eksplorasi penggunaan API lainnya untuk memperkaya fitur *frontend*
 - Implementasikan manajemen kesalahan dan tindak lanjut terkait tanggapan API untuk meningkatkan pengalaman pengguna

Dengan demikian, tutorial ini memberikan landasan kuat bagi pembaca untuk memahami dan menerapkan konsep-konsep *machine learning*, pengembangan *webservice*, integrasi dengan cloud functions, dan penggunaan API dalam konteks praktis. Semua ini bertujuan untuk memberikan wawasan

yang holistik dan praktis bagi pembaca yang ingin memperdalam pemahaman mereka dalam pengembangan solusi teknologi yang melibatkan machine learning.

DAFTAR PUSTAKA

- Ardiansyah, R. (2019). Analisis Sentimen Calon Presiden Dan Wakil Presiden Periode 2019-2024 Pasca Debat Pilpres Di Twitter. *ScientiCO : Computer Science and Informatics Journal*, 2(1).
- Arshad Busro Cahyono, S., Firliana, R., Najibuloh Muzzaki, M., Sari Wardani, A., Iqbal Khalid, M., Wicak Milbar Gamas, A., & Setiawan, H. (2022). *Bulletin of Information Technology (BIT) Rancangan Pembuatan Api Website Data Tanaman Obat Dan Langka Kabupaten Kediri*. 3(4), 255–260.
<https://doi.org/10.47065/bit.v3i1>
- Atmodjo, J. T., Doktoral, M., Komunikasi, J., & Padjajaran, U. (2014). Dinamika Partisipasi Politik melalui Sosial DINAMIKA PARTISIPASI POLITIK REMAJA MELALUI MEDIA SOSIAL. In *Jurnal Visi Komunikasi* (Vol. 13, Issue 02).
- Biswas, P., Hakim, Z., & Hasan, A. (2022). *Development of Chlorophyll-a Soft Sensor Using Machine Learning and IOT*.
<https://doi.org/10.31219/osf.io/v9cy2>
- Buntoro, G. A., Arifin, R., Syaifuddiin, G. N., Selamat, A., Krejcar, O., & Fujita, H. (2021). Implementation of a Machine Learning Algorithm for Sentiment Analysis of Indonesia's 2019 Presidential Election. *IIUM Engineering Journal*, 22(1), 78–92. <https://doi.org/10.31436/IIUMEJ.V22I1.1532>
- Cahyo Santoso, B., Natasya, Y., Willian, S., & Alfando, F. (2020). Tinjauan Pustaka Sistematis terhadap Basis Data MongoDB. *JII: Jurnal Inovasi Informatika Universitas Pradita*, 5(2).
- Christina, S., & Ronaldo, D. (2020). Studi Literatur Sistematis Terhadap Pengembangan Leksikon Sentiment. *Jurnal ELTIKOM*, 4(2), 121–131.
<https://doi.org/10.31961/eltikom.v4i2.211>
- Delvika, B., Abror, N., & Rizky Gurning, U. (2023). *Presidential Election on Twitter Perbandingan Algoritma NBC dan C4.5 Dalam Analisa Sentimen*

Pemilihan Presiden 2024 Pada Twitter.

<https://journal.irpi.or.id/index.php/sentimas>

Fais Sya' bani, M. R., Enri, U., & Padilah, T. N. (2022). Analisis Sentimen Terhadap Bakal Calon Presiden 2024 Dengan Algoritme Naïve Bayes. *JURIKOM (Jurnal Riset Komputer)*, 9(2), 265.

<https://doi.org/10.30865/jurikom.v9i2.3989>

Ferdiana Sari, E. (2021). *Penerapan Github Sebagai Media E-learning Untuk Mengetahui Keefektifan Kolaborasi Project Pada Mata Pelajaran Pemrograman Web Dan Perangkat Bergerak Di Smk Negeri 2 Surabaya* (Vol. 06).

Fernando, D. (2018). *Visualisasi Data Menggunakan Google Data Studio.*

Hadian, N., Hakim, M., & Fanani, M. R. (2023). Implementasi Model Service-Oriented Architecture (SOA) dalam Perancangan Sistem Informasi UMKM. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 5(3), 311–318.

<https://doi.org/10.47233/jteksis.v5i3.879>

Hafidah, P. A., & Frendiana, V. (2023). *Perancangan Aplikasi Mobile Change Request form (CRF) Menggunakan Framework Flutter* (Vol. 2, Issue 1).

Hasan, K., Husna, A., & Fitri, D. (2023). Transformasi Komunikasi Massa Era Digital Antara Peluang Dan Tantangan. *JPP Jurnal Politik Dan Pemerintahan*, 8, 41–55.

Huang, Y., Bian, Y., Li, R., Zhao, J. L., & Shi, P. (2019). Smart contract security: A software lifecycle perspective. In *IEEE Access* (Vol. 7, pp. 150184–150202). Institute of Electrical and Electronics Engineers Inc.

<https://doi.org/10.1109/ACCESS.2019.2946988>

Ishaq, R. el, & Mahanani, P. A. R. (2018). *Media Sosial, Ruang Publik, dan Budaya “POP.”* 3(1).

Jimmy, Hermaliani, E. H., & Kurniawati, L. (2023). Analisis Klasifikasi Sentimen Penggunaan Media Sosial Twitter Terhadap Penundaan Pemilu Presiden

Tahun 2024. *Jurnal Indonesia : Manajemen Informatika Dan Komunikasi*, 4(2), 570–579. <https://doi.org/10.35870/jimik.v4i2.243>

Khotimah, A. C., & Utami, E. (2022). Comparison Naïve Bayes Classifier, K-nearest Neighbor And Support Vector Machine In The Classification Of Individual On Twitter Account. *Jurnal Teknik Informatika (JUTIF)*, 3(3). <https://doi.org/10.20884/1.jutif.2022.3.3.254>

Lavandaia Dharma Bali, Y., Ashril Rizal, A., Puji Indra Kharisma, L., Studi Teknik Informatika, P., Studi Sistem Informasi, P., Syaikh Zainuddin Nahdlatul Wathan, S., Timur, L., & Tenggara Barat, N. (2021). *Peningkatan Efektifitas Programming Dengan Pelatihan Python for Data Science Bagi Komunitas Programming Pondok Pesantren Nahdlatul Wathan Anjani*. 1(1). <http://jurnalwidyalaksmi.com>

Lusita, D., Anissa, F., & Andryani, R. (2022). Penerapan *Cloud computing* Dalam Aplikasi Panggil Teknisi Berbasis Android Menggunakan Google Cloud Platform. In *Jurnal Sains Komputer & Informatika (J-SAKTI)* (Vol. 6, Issue 2).

Mahrozi, N., & Faisal, M. (2023). Analisis Perbandingan Kecepatan Algoritma Selection Sort Dan Bubble Sort. *Jurnal Ilmiah Sain Dan Teknologi*, 1(2), 89–98.

Manullang, O., Prianto, C., & Harani, N. H. (2023). *Analisis Sentimen Untuk Memprediksi Hasil Calon Pemilu Presiden Menggunakan Lexicon Based dan Random Forest*.

Nakhaei, K., Ansari, F., & Ansari, E. (2020). JSSignature: eliminating third-party-hosted JavaScript infection threats using digital signatures. *SN Applied Sciences*, 2(1). <https://doi.org/10.1007/s42452-019-1805-5>

Niar, Y., Komariah, K., Surip, A., Saputra, R., & Ali, I. (2022). Implementasi Algoritma Naïve Bayes Untuk Prediksi Persediaan Barang Rotan. *KOPERTIP : Jurnal Ilmiah Manajemen Informatika Dan Komputer*, 4(1), 28–34. <https://doi.org/10.32485/kopertip.v4i1.112>

- Noorikhsan, F. F., Ramdhani, H., Sirait, B. C., & Khoerunisa, N. (2023). Dinamika Internet, Media Sosial, dan Politik di Era Kontemporer: Tinjauan Relasi Negara-Masyarakat. *Journal of Political Issues*, 5(1), 95–109.
<https://doi.org/10.33019/jpi.v5i1.131>
- Nur, K., Ryadin, F., Saputra, W., Muhammad, A., & Hidayat, N. (2023). *Sistem Pencatatan Kelahiran Dan Kematian Berbasis Cloud computing Kabupaten Jeneponto* (Vol. 1, Issue 1).
- Oktiriani, Q., Kelik Nugroho, A., & Maryanto, E. (2022). Frontend Development in The Final Study Management System (Sipeda) at The Engineering Faculty of Jenderal Soedirman University. *Jurnal Teknik Informatika (JUTIF)* , 3(2), 321–329. <https://doi.org/10.20884/1.jutif.2022.3.2.186>
- Park, J., Ryou, Y., Park, J., & Ryu, S. (2017). Analysis of JavaScript web applications using SAFE 2.0. *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering Companion, ICSE-C 2017*, 59–62. <https://doi.org/10.1109/ICSE-C.2017.4>
- Pasek, P., Mahawardana, O., Sasmita, G. A., Agus, P., & Pratama, E. (2022). Analisis Sentimen Berdasarkan Opini dari Media Sosial Twitter terhadap “Figure Pemimpin” Menggunakan Python. In *JITTER-Jurnal Ilmiah Teknologi dan Komputer* (Vol. 3, Issue 1).
- Perron, M., Castro Fernandez, R., Dewitt, D., & Madden, S. (2020). Starling: A Scalable Query Engine on Cloud Functions. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 131–141. <https://doi.org/10.1145/3318464.3380609>
- Puad, S., & Susilo Yuda Irawan, A. (2023). Analisis Sentimen Masyarakat Pada Twitter Terhadap Pemilihan Umum 2024 Menggunakan Algoritma Naïve Bayes. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 7, Issue 3).
- Rifandi, R., Sutarti, & Anharudin. (2021). Rancang Bangun Kamera Pengawas Menggunakan Raspberry Dengan Aplikasi Telegram Berbasis Internet of Things. *Jurnal PROSISKO*, 8(1). www.amazone.com

- Koneru, A., Bhavani, N. B. N. S. R., Rao, K. P., Prakash, G. S., Kumar, I. P., & Kumar, V. V. (2018). Sentiment Analysis on top five Cloud Service Providers in the Market. *Proceedings of the 2nd International Conference on Trends in Electronics and Informatics (ICOEI 2018)*, 293–297.
- Septia Maharani, A., Fadia Alifa, C., Laila Febriga, A., ZAPagar Alam, J., Ratu, L., Kedaton, K., & Bandar Lampung, K. (2022). Perancangan Data Base Kasir Dan Persediaan Barang Menggunakan Mongodb. In *JDMSI* (Vol. 3, Issue 1).
- Simangunsong, E., & Handoko, R. (2020). The Role of Social Media in Indonesia for Business Transformation Strategy. *International Research Journal of Business Studies*, 8(1), 99–112.
- Singgalen, Y. A. (2023a). Analisis Sentimen Wisatawan terhadap Kualitas Layanan Hotel dan Resort di Lombok Menggunakan SERVQUAL dan CRISP-DM. *Building of Informatics, Technology and Science (BITS)*, 4(4). <https://doi.org/10.47065/bits.v4i4.3199>
- Singgalen, Y. A. (2023b). Penerapan Metode CRISP-DM dalam Klasifikasi Data Ulasan Pengunjung Destinasi Danau Toba Menggunakan Algoritma Naïve Bayes Classifier (NBC) dan Decision Tree (DT). *Jurnal Media Informatika Budidarma*, 7(3), 1551–1562. <https://doi.org/10.30865/mib.v7i3.6461>
- Tineges, R., Triayudi, A., & Sholihat, I. D. (2020). Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi Support Vector Machine (SVM). *Jurnal Media Informatika Budidarma*, 4(3), 650. <https://doi.org/10.30865/mib.v4i3.2181>
- Wahyu Sholeha, E., Yunita, S., Hammad, R., Cahya Hardita, V., Rekayasa Komputer Jaringan, T., & Tanah Laut, P. (2022). Analisis Sentimen Pada Agen Perjalanan Online Menggunakan Naïve Bayes dan K-Nearest Neighbor. *Jurnal Teknologi Informasi Dan Multimedia*, 3(4), 203–208.
- Wulan Ayu, I., & Fitriyanto, S. (2022). Budaya Digital Dalam Transormasi Digital Menghadapi Era Society 5.0. *Jurnal Pengembangan Masyarakat Lokal*, 20–25. <http://e-journallppmunsa.ac.id/index.php/jpml>

Yusuf, L., Hidayatulloh, T., Nurlaela, D., Utami, L. D., & Hasan, F. N. (2020). Simple Additive Weighting untuk Front-end Framework Terbaik. *ILKOM Jurnal Ilmiah*, 12(2), 136–142.
<https://doi.org/10.33096/ilkom.v12i2.593.136-142>

GLOSARIUM

A

Analisis sentiment: proses mengidentifikasi opini yang ada di dalam suatu teks untuk memahami pendapat yang ada di dalamnya.

Akurasi: Ukuran seberapa baik model memprediksi dengan benar pada data yang belum pernah dilihat sebelumnya.

B

Naïve Bayes Classifier : metode penggolongan yang menggunakan teorema bayes pakai telaahan sederhana (naif) bahwa setiap fitur bagian dalam petunjuk merupakan independent antara satu dengan yang lain untuk menggolongkan data kedalam kelas yang sesuai..

C

Confusion Matrix: Tabel yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan membandingkan prediksi model dengan label sebenarnya.

D

Data Latih: Kumpulan data yang digunakan untuk melatih model dalam machine learning. Data ini memiliki label atau hasil yang sudah diketahui.

E

F

Fitur: Representasi dari data yang digunakan dalam model machine learning, seperti kata-kata dalam analisis teks.

G

H

I

J

K

Klasifikasi: proses meletakkan data sesuai kategori data tersebut berdasarkan dari atribut yang ada.

Klasifikasi sentiment: pengelompikan dokumen dalam kategori sentiment, seperti positif, negative, atau netral.

L

M

Machine Learning (ML): Cabang dari kecerdasan buatan yang mengajarkan komputer untuk belajar dari data dan membuat prediksi atau keputusan tanpa secara eksplisit diprogram.

N

Netral: Klasifikasi dalam analisis sentimen yang menunjukkan bahwa teks atau pendapat tidak memiliki polaritas positif atau negatif yang kuat.

O

Overfitting: Keadaan di mana model machine learning terlalu dipasangkan pada data latih dan tidak bisa melakukan generalisasi dengan baik pada data baru.

P

Pemilihan Fitur: Proses memilih fitur yang paling informatif atau relevan untuk meningkatkan kinerja model.

Pemrosesan Teks: Proses pengolahan data teks untuk mengubahnya menjadi format yang dapat dipahami oleh mesin, termasuk tahap pembersihan, tokenisasi, dan vektorisasi.

Preprocessing: analisis tahap awal atau teks mentah sehingga menjadi format yang terstruktur.

Probabilitas kondisional: kemungkinan yang terjadi pada suatu peristiwa, diberikan bahwa peristiwa lain telah terjadi.

Probabilitas Bersyarat: Probabilitas suatu kejadian yang terjadi dengan mempertimbangkan terjadinya kejadian lain.

Probabilitas Marginal: Probabilitas suatu kejadian yang dihitung tanpa memperhatikan atau mempertimbangkan kejadian lain.

Q

R

S

Sentimen: Emosi, opini, atau sikap yang terkandung dalam suatu teks atau dokumen.

T

Text Mining: Proses ekstraksi informasi penting dari teks yang tidak terstruktur untuk mendapatkan wawasan yang bermanfaat.

Teorema bayes: rumus yang berguna untuk menghitung peluang tentative dari 2 peristiwa yang saling berkaitan.

Text mining: proses pendataan informasi yang berguna dari teks yang tidak beraturan

Tokenisasi: Proses memecah teks menjadi token-token (kata-kata, frasa, atau karakter) agar lebih mudah dianalisis.

U

Underfitting: Keadaan di mana model machine learning terlalu sederhana untuk menggambarkan kompleksitas data dan gagal dalam memahami pola yang ada.

V

Validasi model: proses mengevaluasi untuk memastikan bahwa model yang digunakan memberikan prediksi yang akurat serta dapat diterapkan ke data baru

Validasi Silang (Cross-validation): Teknik untuk menguji kinerja model dengan membagi data menjadi subset yang saling terpisah untuk melatih dan menguji model.

Vektorisasi: Konversi teks menjadi representasi vektor numerik untuk digunakan dalam model machine learning.

W



X



Y



KREDIT GAMBAR

https://www.python.org/static/community_logos/python-logo-master-v3-TM.png

https://upload.wikimedia.org/wikipedia/commons/3/38/Jupyter_logo.svg

https://en.m.wikipedia.org/wiki/File:Visual_Studio_Code_1.35_icon.svg

<https://voyager.postman.com/logo/postman-logo-orange.svg>

<https://logos-world.net/github-logo/>

https://upload.wikimedia.org/wikipedia/commons/5/51/Google_Cloud_logo.svg

https://go.dev/blog/go-brand/Go-Logo/PNG/Go-Logo_Blue.png

<https://upload.wikimedia.org/wikipedia/commons/6/6a/JavaScript-logo.png>

https://upload.wikimedia.org/wikipedia/commons/9/93/MongoDB_Logo.svg

TENTANG PENULIS



Dani Ferdinand. Lahir di Kebumen, 9 November 2002. Saat ini adalah mahasiswa Semester 5 Program Studi D4 Teknik Informatika di Universitas Logistik dan Bisnis Internasional.



Aryka Anisa Pertiwi. Lahir di Bantul, 22 April 2003. Saat ini adalah mahasiswa Semester 5 Program Studi D4 Teknik Informatika di Universitas Logistik dan Bisnis Internasional.



Nisa Hanum Harani adalah seorang sarjana yang telah menyelesaikan studinya pada S1 Sistem Informasi di Universitas Maranatha dan S2 Sistem Informasi di Institut Teknologi Bandung. Beliau memiliki kegiatan utama sebagai Dosen Teknik Informatika pada Universitas Logistik dan Bisnis Internasional.

MENGENAL MACHINE LEARNING

ANALISIS SENTIMEN MENGGUNAKAN NAIVE BAYES CLASSIFIER

Buku tutorial ini menjelajahi dampak era digital terhadap interaksi sosial masyarakat Indonesia melalui media sosial. Fokus utama adalah pentingnya akses internet dan peran aktif masyarakat dalam menciptaka konten. Buku ini membahas penelitian sentimen masyarakat dengan menggunakan data komentar media sosial, dengan studi kasus pada pemilihan presiden 2024. Teknologi berperan sebagai kunci, termasuk pendekatan webservice, cloud computing, dan pemodelan sentimen menggunakan algoritma Naive Bayes Classifier dengan python. frontend dibangun dengan HTML, CSS (framework bulma), dan JavaScript Es6, sementara backend menggunakan Golang dan Google Cloud Function. Buku ini tidak hanya memberikan wawasan analisis sentimen, tapi juga menjadi panduan praktiss dalam menerapkan teknologi terkini pada era digital .