

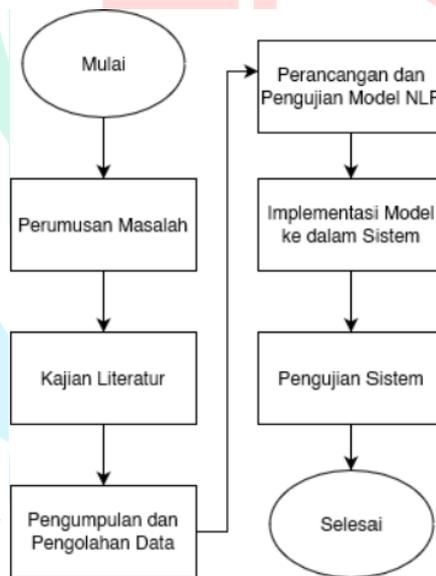
BAB III

TAHAPAN PELAKSANAAN

Dalam penelitian ini tahap pelaksanaan terdiri dari langkah-langkah pelaksanaan, metode pengembangan sistem dan metode pengujian.

3.1 Langkah-langkah Pelaksanaan

Sebuah flowchart langkah pelaksanaan penelitian adalah suatu alat visual yang digunakan untuk menggambarkan penjelasan dan alur penelitian yang akan dilakukan oleh seorang peneliti dalam rangka menyelesaikan tugas akhirnya. Flowchart ini terdiri dari serangkaian tahapan yang diikuti peneliti dalam menjalankan penelitian, dan membantu dalam memahami dan mengkomunikasikan langkah-langkah yang diperlukan dalam penelitian tersebut.



Gambar 3. 1 Flowchart Penelitian

1. Perumusan masalah

Tahap pertama dalam penelitian ini adalah mengidentifikasi masalah. Masalah ini kemudian diformulasikan untuk memfasilitasi pencarian solusi dalam penelitian ini. Peneliti melakukan observasi

terkait isu subsidi kendaraan listrik dan menentukan sumber data yang akan digunakan.

2. Kajian literatur

Dalam tahap kajian literatur penelitian ini, peneliti melakukan studi terhadap sumber bacaan yang telah dipublikasikan yang terkait dengan analisis sentimen, arsitektur LSTM, dan isu subsidi kendaraan listrik. Sumber-sumber yang digunakan untuk mendukung penelitian ini meliputi jurnal, buku, laporan studi, penelitian sebelumnya, serta dokumen-dokumen lain yang relevan.

3. Pengumpulan data dan pengolahan data

Penelitian ini menggunakan data opini masyarakat terkait subsidi kendaraan listrik. Data sampel yang digunakan adalah data pada akhir tahun 2022 dengan rentang waktu dari 1 Desember hingga 1 Agustus 2023.

Sumber data yang dikumpulkan berasal dari media sosial Twitter, dan proses pengumpulan data dilakukan dengan menggunakan metode scraping API Twitter. Metode scraping memungkinkan pengumpulan data dalam jumlah besar dalam waktu yang relatif singkat dibandingkan dengan pengumpulan data manual. Data yang diambil merupakan tweet yang mengandung kata kunci tertentu dan telah difilter untuk bahasa Indonesia.

Setelah pengumpulan data, data tersebut kemudian masuk ke tahap pra-proses data. Tujuan dari tahap ini adalah membersihkan data yang awalnya kotor menjadi data yang bersih dan siap untuk digunakan dalam analisis penelitian.

4. Perancangan dan pengujian model

Perancangan model NLP adalah komponen kunci dalam penelitian ini. Model NLP dibangun dengan menggunakan framework TensorFlow dan mengadopsi arsitektur LSTM. Model tersebut

kemudian disimpan dalam format dengan ekstensi h5. Peran utama dari model ini adalah untuk mengatasi permasalahan yang diajukan dalam penelitian ini, yaitu menentukan opini publik berdasarkan masukan yang diberikan oleh pengguna. Model NLP ini bertugas untuk menganalisis dan menginterpretasikan opini-opini tersebut.

5. Implementasi model ke dalam sistem

Tahap implementasi ini melibatkan pembuatan aplikasi sesuai dengan rancangan yang telah disusun. *Framework* yang digunakan dalam pengembangan aplikasi web ini adalah Flask, dan bahasa pemrograman yang digunakan adalah Python. Implementasi aplikasi ini terdiri dari dua sisi utama, yaitu antarmuka pengguna (*front-end*) dan logika pengolahan data (*back-end*).

Sisi front-end bertanggung jawab atas tampilan aplikasi yang dilihat oleh pengguna. Ini termasuk tampilan grafis, elemen antarmuka pengguna, serta cara pengguna berinteraksi dengan aplikasi.

Sisi back-end mengacu pada logika pengolahan data dan fungsionalitas aplikasi. Di sini, model NLP yang telah dilatih dan tokenizer yang diperoleh dari dataset digunakan untuk menganalisis dan mengolah data yang masuk. Back-end juga bertanggung jawab untuk memberikan hasil analisis kepada pengguna melalui antarmuka pengguna.

Kombinasi antara *front-end* dan *back-end* menciptakan aplikasi lengkap yang memungkinkan pengguna untuk berinteraksi dengan model NLP dan mendapatkan hasil analisis terkait dengan opini publik berdasarkan masukan yang mereka berikan.

6. Pengujian sistem

Sistem yang telah dikembangkan akan mengalami serangkaian pengujian untuk meminimalisasi kesalahan program atau bug. Pengujian sistem akan menggunakan metode *black box* dan *white box*.

Metode *black box* dipilih dengan tujuan untuk memastikan bahwa aplikasi bekerja dengan baik dan sesuai dengan fungsinya. Dalam pengujian *black box*, fokus utama adalah pada input dan output aplikasi, tanpa mempertimbangkan rincian internal kode. Hal ini membantu memastikan bahwa aplikasi dapat digunakan sesuai dengan harapan pengguna.

Metode *white box* dipilih untuk memastikan bahwa kode yang telah ditulis berjalan dengan baik. Dalam pengujian *white box*, fokus utama adalah pada struktur internal kode, serta bagaimana kode tersebut dieksekusi. Hal ini membantu memastikan bahwa setiap bagian dari kode berfungsi dengan benar dan efisien.

Kombinasi antara pengujian *black box* dan *white box* membantu memastikan kualitas dan keandalan sistem yang dikembangkan, serta meminimalisasi kemungkinan kesalahan atau bug dalam aplikasi.

3.2 Metode Pengembangan

Pengembangan perangkat lunak memiliki beberapa metode. Peneliti menggunakan metode purwarupa dalam mengembangkan perangkat lunak. Metode purwarupa (*prototype*) adalah pendekatan dalam pengembangan produk atau perangkat lunak yang melibatkan pembuatan model awal atau versi sederhana dari sistem yang akan dikembangkan. Tujuan dari metode ini adalah untuk menguji ide, konsep, atau desain sebelum implementasi penuh. Berikut adalah beberapa aspek dari metode purwarupa:

- **Pembuatan Model Awal:** Metode purwarupa melibatkan pembuatan versi awal dari produk atau sistem, yang dikenal sebagai purwarupa. Model ini bisa berupa sketsa, mockup, atau versi fungsional sederhana dari sistem akhir.
- **Uji Coba dan Umpan Balik:** Purwarupa digunakan untuk mengumpulkan umpan balik dari pengguna atau stakeholder. Dengan menguji model awal ini, pengembang dapat

mengidentifikasi masalah, kebutuhan, dan preferensi pengguna sebelum mengembangkan sistem akhir.

- Iterasi dan Penyempurnaan: Proses pembuatan purwarupa biasanya bersifat iteratif. Berdasarkan umpan balik yang diterima, purwarupa diperbaiki dan disempurnakan dalam beberapa iterasi hingga mencapai desain yang diinginkan.
- Pengurangan Risiko: Dengan mengidentifikasi dan mengatasi masalah pada tahap awal, metode purwarupa membantu mengurangi risiko dan potensi kegagalan dalam pengembangan produk akhir.
- Komunikasi: Purwarupa juga berfungsi sebagai alat komunikasi yang efektif antara tim pengembang dan stakeholder, membantu untuk memastikan bahwa kebutuhan dan ekspektasi dipahami dengan jelas.
- Penghematan Biaya dan Waktu: Meskipun pembuatan purwarupa memerlukan investasi awal, ini dapat menghemat biaya dan waktu dalam jangka panjang dengan mencegah perubahan besar yang mahal pada tahap akhir pengembangan.

Secara keseluruhan, metode purwarupa adalah pendekatan yang berguna untuk merancang dan mengembangkan sistem atau produk dengan lebih efisien, memastikan bahwa solusi akhir lebih sesuai dengan kebutuhan dan harapan pengguna.

3.3 Metode Pengumpulan Data

Pada tahap pengumpulan data, penulis mencari dan menyimpan informasi yang dibutuhkan dalam pembangunan sistem. Pengumpulan data dilakukan dengan Teknik *scrapping*. Pengambilan data menggunakan scraping dengan API adalah metode yang efisien dan terstruktur, di mana aplikasi mengirim permintaan ke endpoint API untuk mengakses dan mengunduh data yang diperlukan dari server, memungkinkan peneliti untuk mendapatkan informasi yang relevan dengan cara yang lebih cepat dan terorganisir dibandingkan dengan scraping tradisional dari halaman web.

3.3 Metode Pengujian

Metode pengujian memang berperan penting dalam memastikan kualitas dan kinerja sistem atau produk yang dikembangkan. Beberapa manfaat utama dari metode pengujian meliputi:

1. Memastikan Kinerja yang Baik: Metode pengujian membantu memastikan bahwa sistem atau produk berjalan sesuai dengan spesifikasi dan berfungsi dengan baik sesuai dengan tujuannya.
2. Evaluasi Kualitas: Pengujian memberikan kesempatan untuk mengevaluasi kualitas produk atau sistem, yang mencakup aspek-aspek seperti keandalan, kecepatan, keamanan, dan lain-lain.
3. Pendeteksian Kesalahan: Salah satu manfaat utama dari pengujian adalah kemampuannya untuk mendeteksi kesalahan atau bug dalam sistem sebelum produk tersebut dirilis ke pengguna akhir. Ini membantu dalam mengurangi risiko masalah yang mungkin muncul di kemudian hari.
4. Validasi Persyaratan: Metode pengujian membantu memvalidasi bahwa sistem atau produk memenuhi persyaratan dan harapan yang telah ditetapkan.

Dalam penelitian ini metode pengujian yang digunakan adalah *black box* dan *white box*, yang merupakan pendekatan yang umum digunakan untuk memeriksa fungsionalitas dan integritas perangkat lunak. Kombinasi kedua metode ini dapat membantu memastikan bahwa aplikasi analisis sentimen yang dikembangkan bekerja dengan baik, dan meminimalisasi risiko kesalahan atau bug sebelum aplikasi tersebut dirilis kepada pengguna akhir.

3.4.1 Black Box

Pengujian *black box* berfokus pada fungsionalitas sistem dari perspektif pengguna atau sisi depan tanpa memeriksa detail kode atau alur data yang ada di dalamnya.

Pada pengujian *black box*, setiap fungsi atau fitur yang ada di antarmuka pengguna (*front-end*) diuji dengan cara menekan tombol, memberikan input pada formulir, mengunggah berkas, serta mengamati hasil keluaran dan tampilan aplikasi. Tujuannya adalah memastikan bahwa aplikasi berperilaku sesuai dengan harapan pengguna dan dapat digunakan dengan baik.

Pengujian *black box* adalah metode yang sangat berguna untuk memastikan bahwa pengguna akhir akan memiliki pengalaman yang baik dengan aplikasi, dan masalah atau kesalahan yang mungkin muncul dapat diidentifikasi dan diperbaiki sebelum aplikasi dirilis.

3.3.1 *White Box*

- Pengujian *white box* memang difokuskan pada pemeriksaan struktur kode dalam tahap perancangan. Tujuan utamanya adalah memastikan bahwa kode berfungsi dengan baik dan bahwa proses input dan output kode bekerja sesuai yang diharapkan.

Salah satu teknik yang digunakan adalah *statement coverage*, Teknik tersebut melibatkan pengujian minimal satu kali terhadap setiap baris kode. Ini berarti setiap baris kode harus dieksekusi selama pengujian untuk memastikan bahwa tidak ada bagian dari kode yang tidak diuji. Dengan cara ini, kesalahan atau potensi *bug* dalam kode dapat diidentifikasi dan diperbaiki.

Pengujian *white box* memiliki peran penting dalam memastikan kualitas kode dan mengurangi risiko kesalahan dalam perangkat lunak. Ini membantu memastikan bahwa setiap bagian dari kode berjalan dengan baik dan sesuai dengan perancangan yang telah dibuat.