BAB 3
METODOLOGI PENELITIAN

Metode penelitian ini dirancang untuk mengembangkan metode deteksi kesalahan kapitalisasi huruf sesuai Ejaan Yang Disempurnakan (EYD) dengan memanfaatkan algoritma Random Forest.

3.1 Tinjauan Pustaka

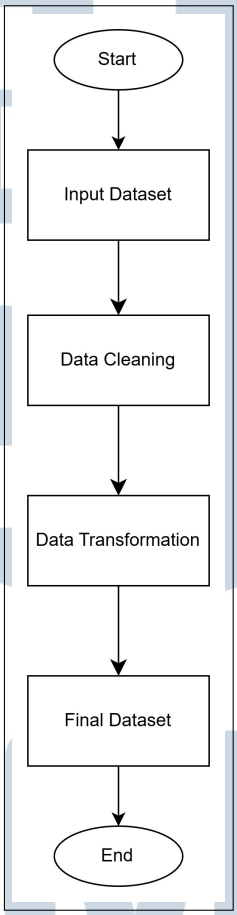
Pada tahap ini, dilakukan tinjauan pustaka berupa riset teoritis dari berbagai sumber seperti jurnal ilmiah, buku, dan artikel penelitian yang berkaitan dengan deteksi kesalahan kapitalisasi huruf. Random Forest dipelajari karena kemampuannya dalam mengklasifikasikan data teks dengan akurasi tinggi, sedangkan pemahaman tentang aturan Ejaan Yang Disempurnakan (EYD) lebih lanjut dieksplorasi untuk meningkatkan kinerja sistem dalam mendeteksi kesalahan kapitalisasi yang sesuai.

3.2 Pengumpulan Data

Pengumpulan data dilakukan dengan mengolah teks berita Tribun yang disimpan di Google Drive. Data yang telah terkumpul kemudian diubah dari format Word atau PDF ke dalam format .txt untuk mempermudah pengelolaan. Selanjutnya, data tersebut diproses lebih lanjut dan dikonversi ke dalam format .csv sebagai bentuk akhir yang siap digunakan pada tahap analisis dan pemrosesan. Selain itu, terdapat dataset tambahan antaranya dataset gelar yang bersumber dari [33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 33], dataset suku, dataset mengenai 25 peristiwa bersejarah di Indonesia yang dirujuk dari [53], dataset nama jenis yang diacu dari [54], dan dataset geografi yg berisi daftar nama kecamatan dan kelurahan di berbagai daerah Jakarta yang dapat ditemukan pada sumber-sumber berikut, Jakarta Pusat [55], Jakarta Utara [56], Jakarta Timur [57], Jakarta Selatan [58], Jakarta Barat [59], Kepulauan Seribu [60], negara-negara di Asia Tenggara yang diambil dari [61],

3.3 Pre-processing

Tahapan preprocessing dalam penelitian ini mencakup beberapa langkah utama untuk memastikan data siap digunakan dalam proses analisis dan pelatihan model. Tahapan tersebut terlampir pada Gambar 3.1 dibawah ini:



Gambar 3.1. Preproccesing Flowchart

Proses preprocessing diawali dengan konversi format file, di mana data Tribun yang awalnya berformat Word (.docx) diubah menjadi format .txt untuk mempermudah pengolahan. Selanjutnya, dataset disimpan dalam format .csv untuk analisis lebih lanjut. Tahap berikutnya adalah pembersihan data (*data cleaning*) yang mencakup penghapusan elemen-elemen tidak relevan, seperti tautan, domain web (contoh: example.com), karakter khusus seperti `@` dan `#`, simbol aneh, tanda kutip berlebih, serta strip tunggal (`-`) atau ganda (`--`). Spasi berlebih, baris kosong, tanggal, atau baris dengan satu kata juga dihapus. Selain itu, pola header berita seperti `TRIBUNNEWS.COM, JAKARTA -` juga dihapus, bersama dengan baris

duplikat berdasarkan kolom *Sentence* dan baris yang memiliki nilai `NaN` pada kolom relevan. Dataset yang telah dibersihkan kemudian disimpan dalam file `.csv` baru.

Tahap terakhir adalah pengolahan dataset, di mana kolom baru ditambahkan untuk berbagai transformasi teks. Kolom tersebut mencakup versi teks pada kolom *Sentence* yang diubah menjadi huruf kecil seluruhnya (*lowercase*), versi dengan kombinasi huruf besar dan kecil secara acak (*randomcase*), serta versi teks tanpa tanda baca (*nopunctuation*). Dataset yang telah diproses ini kemudian disiapkan untuk proses ekstraksi fitur EYD, pelabelan, dan tokenisasi.

3.4 Ekstraksi Fitur

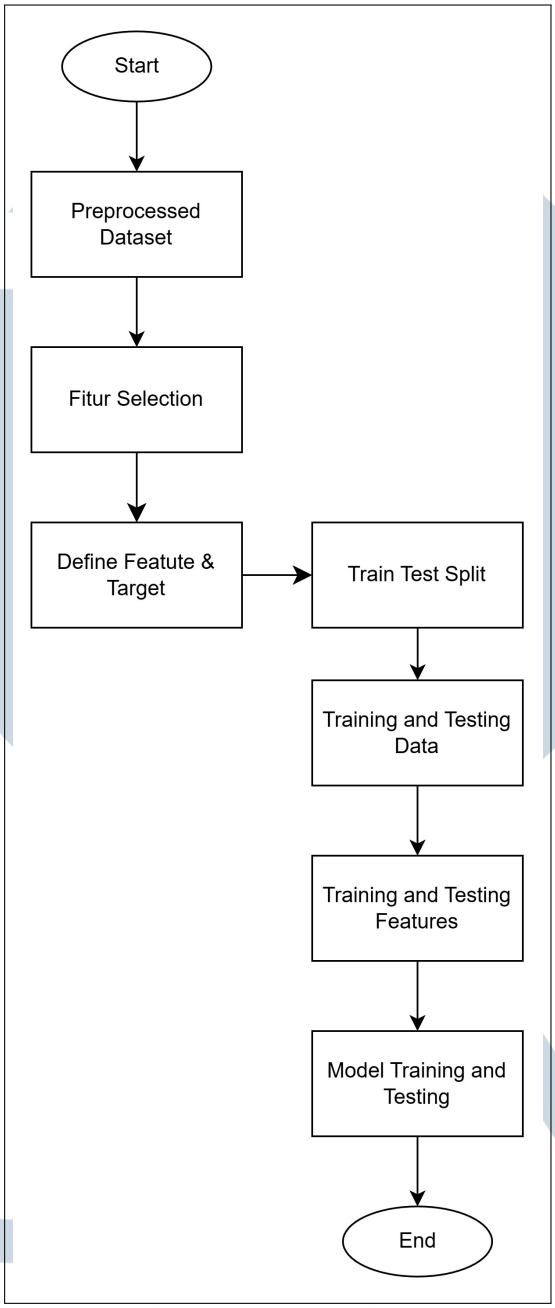
Tahapan ekstraksi fitur dalam penelitian ini bertujuan untuk mengambil informasi yang relevan dari data teks sehingga dapat digunakan sebagai input untuk algoritma Random Forest.

Ekstraksi fitur dilakukan berdasarkan pendekatan berbasis aturan EYD 5 dan konteks teks, yang masing-masing memberikan kontribusi berbeda terhadap proses deteksi kesalahan kapitalisasi. Proses ini mencakup analisis kata berdasarkan aturan kapitalisasi EYD 5 untuk menentukan apakah sebuah kata mematuhi atau melanggar aturan yang telah ditentukan.

Sebagai bagian dari proses ini, langkah pelabelan dan tokenisasi juga menjadi elemen penting untuk mendukung ekstraksi fitur. Proses pelabelan dilakukan dengan menandai setiap kata dalam dataset sesuai dengan kepatuhannya terhadap aturan kapitalisasi EYD. Label diberikan untuk mengidentifikasi kata yang benar (0) atau salah (1). Selain itu, tokenisasi dilakukan untuk membagi teks menjadi unit-unit yang lebih kecil seperti kata atau kalimat, yang kemudian diproses lebih lanjut dalam analisis.

3.5 Pembagian Dataset

Tahapan pembagian dataset (*split dataset*) dilakukan untuk mempersiapkan data yang akan digunakan dalam proses pelatihan dan pengujian model. Tahapan tersebut terlampir pada Gambar 3.2 dibawah ini:

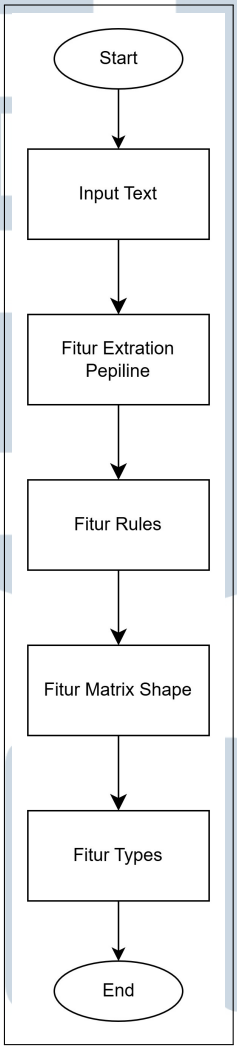


Gambar 3.2. Ekstrasi Fitur Flowchart

Dataset yang telah melalui tahap *preprocessing* dibagi menjadi dua bagian utama, yaitu data latih (*training data*) dan data uji (*testing data*). Data latih digunakan untuk melatih model dalam mengenali pola dan aturan yang sesuai dengan tugas deteksi huruf kapital berdasarkan aturan EYD, sedangkan data uji digunakan untuk mengevaluasi performa model yang telah dilatih. Pembagian dataset ini umumnya dilakukan secara proporsional, seperti 80% .

3.6 Pembangunan Model

Pembangunan model dilakukan dengan menggunakan algoritma Random Forest untuk mendeteksi kesalahan kapitalisasi berdasarkan aturan EYD. Tahapan tersebut terlampir pada Gambar 3.3 dibawah ini:



Gambar 3.3. Pembangunan Model Flowchart

Model ini dilatih menggunakan data latih yang telah melalui tahap *preprocessing* dan pembagian dataset. Algoritma Random Forest dipilih karena kemampuannya dalam menangani data dengan banyak fitur dan menghasilkan model yang dapat melakukan generalisasi dengan baik. Selama pelatihan, model akan mempelajari pola-pola kapitalisasi yang benar dan salah sesuai dengan aturan yang telah ditetapkan. Setelah pelatihan selesai, model siap untuk diuji menggunakan data uji untuk mengevaluasi performanya.

3.7 Evaluasi Model

Setelah model dilatih, tahap selanjutnya adalah evaluasi untuk mengukur performa model dalam mendeteksi kesalahan kapitalisasi. Evaluasi dilakukan dengan menggunakan data uji yang belum pernah dilihat oleh model sebelumnya. Beberapa metrik yang digunakan untuk menilai performa model antara lain *accuracy*, *precision*, *recall*, dan *F1-score*. *Accuracy* mengukur persentase prediksi yang benar, sedangkan *precision* mengukur ketepatan model dalam mengklasifikasikan kesalahan kapitalisasi yang benar. *Recall* menunjukkan seberapa banyak kesalahan kapitalisasi yang berhasil ditemukan dari seluruh kesalahan yang ada, dan *F1-score* adalah kombinasi antara *precision* dan *recall* yang digunakan untuk mengukur keseimbangan antara keduanya. Berdasarkan hasil evaluasi ini, performa model akan dianalisis dan langkah-langkah perbaikan dapat dilakukan untuk meningkatkan akurasi deteksi kesalahan kapitalisasi.