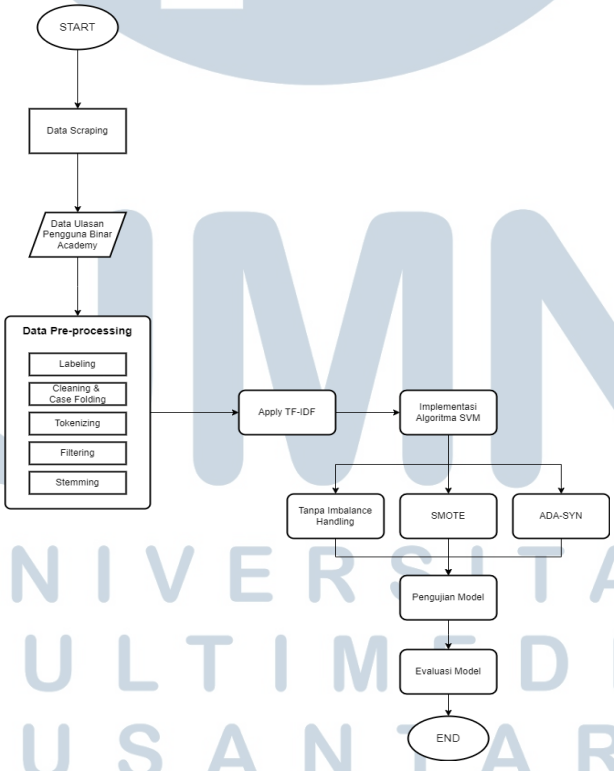
BAB 3
METODOLOGI PENELITIAN

Pengerjaan penelitian ini dilakukan menggunakan data yang berasal dari *google play store*. Data yang didapatkan dari hasil *crawling* menggunakan *google-play scraper* diproses terlebih dahulu pada tahap *pre-processing* agar data siap diolah dan dianalisis lebih lanjut. Berikutnya data dibagi kedalam data *training* dan data *testing* dan dilakukan pembobotan menggunakan TF-IDF. Kemudian, pemodelan dengan menggunakan algoritma *Support Vector Machine* akan dilakukan dengan menggunnakan kernel linear. Data akan diuji menggunakan tiga perbandingan dataset yang berbeda dan dengan pengaplikasian *imbalance handling* seperti SMOTE dann ADA-SYN. Terakhir, model akan dievaluasi menggunakan *confusion matrix* dan divalidasi dengan menggunakan *K-Fold Cross Validation*. Keseluruhan proses yang dilakukan pada pengerjaan penelitian ini dapat dilihat pada gambar 3.1.



Gambar 3.1. Flowchart pengerjaan penelitian

3.1 Data Scraping

Data Sentimen pengguna aplikasi *binar academy* dikumpulkan melalui data ulasan aplikasi pada *Google Play Store*. *Scraping* data ulasan dari *Google Playstore* dilakukan menggunakan *library google-play-scraper* [30]. Data yang diambil menggunakan *library* adalah data berbahasa Indonesia dengan rating 1-5. Data yang diambil disortir berdasarkan data ulasan yang paling relevan. Data yang telah dikumpulkan tersebut kemudian dimasukkan kedalam dataframe dan dijadikan sebuah file dengan format csv. Hal ini diperlukan agar data dapat diproses lebih lanjut dengan sesuai kebutuhan penelitian.

3.2 Data Pre-processing

Gambar 3.2. Flowchart data pre-processing

Data *preprocessing* merupakan tahapan untuk mempersiapkan data sebelum implementasi algoritma agar data siap diproses. Hal ini diperlukan untuk menghindari kesalahan pada model yang akan dibuat. Pada tahap ini, data diolah dalam beberapa proses agar data siap untuk digunakan untuk analisis sentimen menggunakan algoritma *Support Vector Machine* (SVM). Adapun tahapan-tahapan pada proses Data *Pre-processing* adalah *labeling*, *cleaning text dan case folding*, *tokenizing*, *filtering*, dan *stemming*.

3.2.1 Labeling

Labeling merupakan proses untuk memberikan label untuk dapat diklasifikasikan berdasarkan kebutuhan penelitian. Pada penelitian ini label dari sentimen dibagi ke dalam tiga kategori yaitu positif, netral dan negatif. Pelabelan dilakukan dengan menggunakan *score* dari ulasan pengguna.

Tabel 3.1. Contoh *labeling* sentimen

Ulasan	Score	Sentimen
Aplikasi Binar bagus banget, sangat membantu	5	Positive
Sekarang Binar ada aplikasi mobilenya	3	Neutral
Aplikasinya ga bisa dibuka	1	Negative

3.2.2 Cleaning dan Case Folding

Cleaning Text dan *Case Folding*, merupakan proses untuk membersihkan data teks dari simbol-simbol dan karakter tanda baca yang tidak perlu digunakan. Selain itu, semua huruf pada kalimat diubah menjadi huruf kecil agar data yang diolah lebih konsiten.

Tabel 3.2. Contoh *cleaning* dan *case folding*

Ulasan	Cleaning dan Case Folding
Cara baru belajar semua ilmu Digital, sangat membantu	cara baru belajar semua ilmu digital sangat membantu
Sekarang Binar ada aplikasi mobilenya	sekarang binar ada aplikasi mobilenya

3.2.3 Tokenizing

Tokenizing merupakan proses untuk memecah sekumpulan karakter pada teks menjadi satuan kata. Hal ini dilakukan untuk mempermudah pemrosesan teks dan juga mengidentifikasi fitur pada dokumen. Kalimat akan dipecah menjadi kata-kata terpisah.

Tabel 3.3. Contoh *tokenizing*

Ulasan	Tokenizing
cara baru belajar semua ilmu digital sangat membantu	[cara, baru, belajar, semua, ilmu, digital, sangat, membantu]
sekarang binar ada aplikasi mobilenya	[sekarang, binar, ada, aplikasi, mobilenya]

3.2.4 Filtering

Filtering merupakan merupakan proses untuk menghilangkan kata umum yang sering muncul tetapi tidak memiliki arti. Hal ini dilakukan agar isi dari dokumen tidak terkontaminasi dengan isi yang dianggap tidak memiliki makna.

Tabel 3.4. Contoh *filtering*

Ulasan	**filtering**
cara baru belajar semua ilmu digital sangat membantu	[belajar, ilmu, digital, sangat, membantu]
sekarang binar ada aplikasi mobilenya	[binar, aplikasi, mobilenya]

3.2.5 Stemming

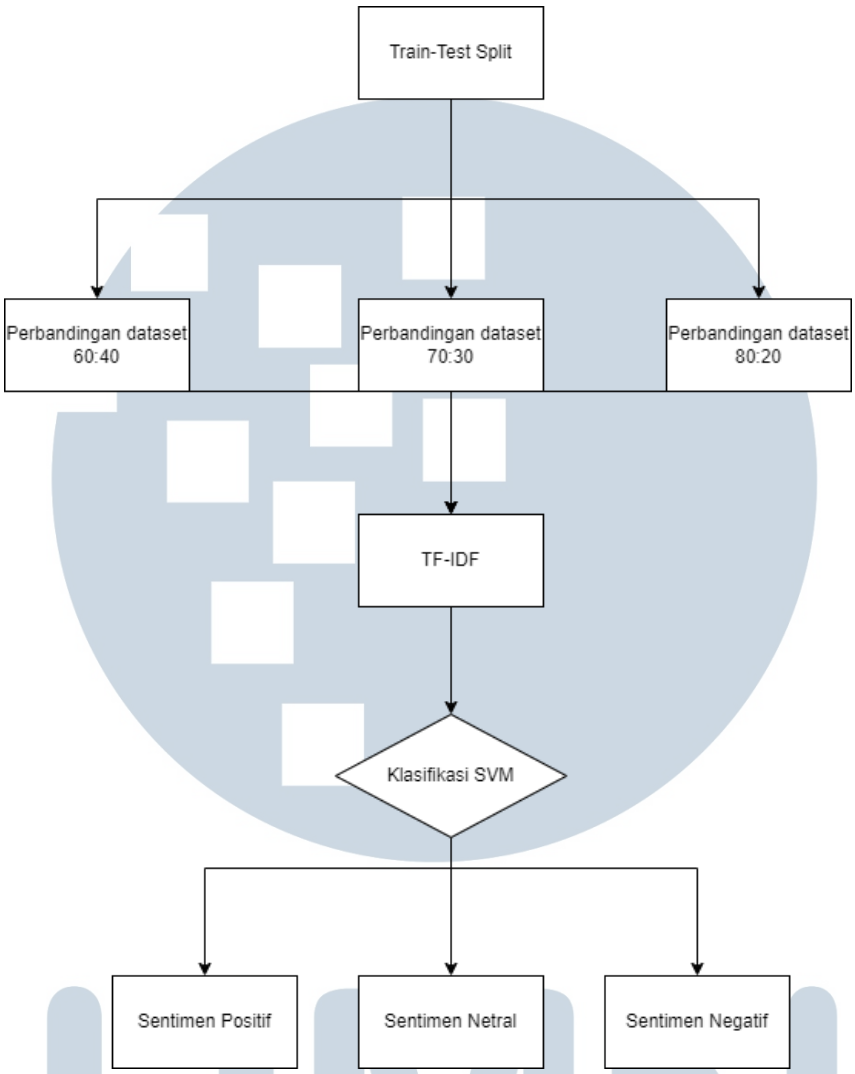
Stemming merupakan proses untuk mereduksi bentuk kata menjadi kata dasar. Hal ini merupakan proses terakhir pada pre-processing sebelum data kembali disatukan kembali menjadi satu kesatuan.

Tabel 3.5. Contoh *stemming*

Ulasan	**stemming**
cara baru belajar semua ilmu digital sangat membantu	belajar ilmu digital sangat bantu
sekarang binar ada aplikasi mobilenya	binar aplikasi mobile

3.3 Pembuatan model

Untuk pembuatan model, data yang telah diproses sebelumnya dibagi kedalam beberapa perbandingan dataset dan dibobotkan dengan TF-IDF. Setelah itu, pengujian dilakukan untuk mengatasi ketidakseimbangan pada data dengan menggunakan SMOTE dan ADA-SYN. Kemudian, algoritma *Support Vector Machine* diterapkan pada model untuk melakukan klasifikasi sentimen positif, negatif dan netral. Alur pembuatan model dapat dilihat pada gambar 3.3.



Gambar 3.3. Pembuatan Model

3.3.1 Train Test Split dan TF-IDF

Setelah melewati data *preprocessing* untuk siap diolah dan dimodelkan, data digabungkan kembali menjadi satu variabel *string*. Data kemudian akan dibagi menggunakan *train test split* untuk dapat dimodelkan dan diuji dengan menggunakan tiga perbandingan rasio dataset yang berbeda. Pengujian data akan dilakukan sebanyak tiga kali yaitu dengan perbandingan dataset 60:40, perbandingan dataset 70:30, dan perbandingan dataset 80:20. Setelah itu, Proses TF-IDF *(Term Frequency-Inverse Document Frequency)* kemudian diterapkan untuk memberi bobot pada kata-kata yang muncul dalam ulasan aplikasi. Dengan TF-IDF, informasi dapat diekstraksi, sehingga model mampu melakukan

pemrosesan yang lebih akurat.

3.3.2 Imbalance Handling

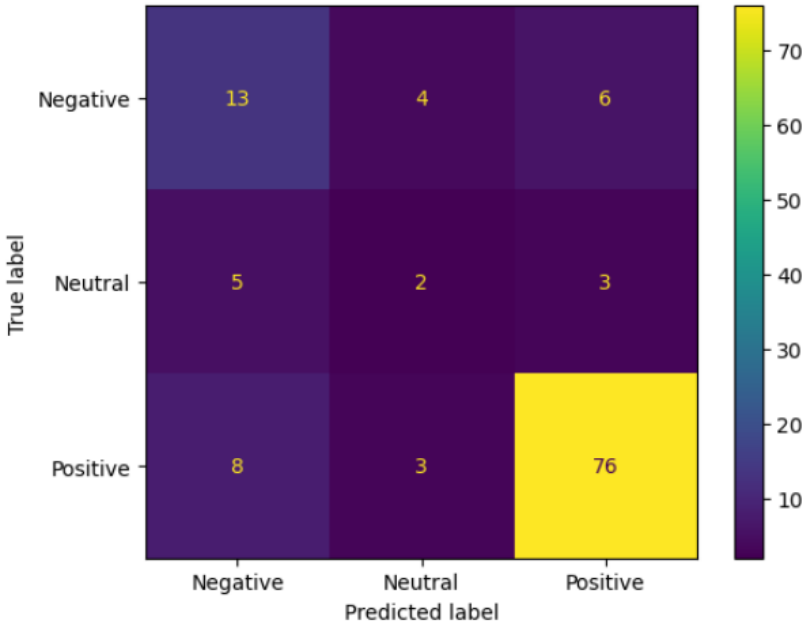
Untuk dapat mengatasi ketidakseimbangan pada dataset dimana jumlah kelas mayoritas tidak seimbang dengan kelas minoritas, diaplikasikan metode *imbalance handling* dengan menggunakan SMOTE dan ADA-SYN. Model yang telah dibangun kemudian akan dibandingkan yaitu dengan data tanpa *imbalance handling*, menggunakan SMOTE dan menggunakan ADA-SYN.

3.3.3 Support Vector Machine

Setelah dilakukan pembagian dataset dan menggunakan metode *imbalance handling*, kemudian pemodelan menggunakan algoritma *Support Vector Machine* dilakukan. Pemodelan SVM dilakukan dengan menggunakan kernel linear untuk melakukan analisis sentimen ulasan pengguna aplikasi *Binar Academy*. Hasil dari klasifikasi yang dilakukan adalah untuk mengelompokkan data kedalam sentimen positif, negatif dan netral.

3.4 Evaluasi Model

Pada tahap evaluasi, pengukuran akan dilakukan menggunakan *Confusion Matrix* untuk mengevaluasi kemampuan algoritma *Support Vector Machine* untuk melakukan klasifikasi data. Pengujian dilakukan dengan menggunakan tabel positif dan negatif. Tabel tersebut memiliki nilai *True*, yang berarti label asli dan memprediksi label memprediksi sentimen yang sama untuk teks, dan nilai *False*, yang berarti label prediksi memprediksi sentimen untuk teks yang berbeda dengan label asli. *Accuracy*, *Precision*, *Recall*, dan *F1-score* akan dihitung dengan menggunakan hasil dari *Confusion Matrix*. Setelah itu *K-Fold Cross Validation* dilakukan untuk memastikan bahwa seluruh dataset telah digunakan untuk pelatihan dan pengujian. Hasil dari *train test split* kemudian akan dibandingkan dengan hasil dari *K-Fold Cross Validation*. Hasil dari *k-fold cross validation* umumnya memiliki hasil yang lebih baik dibanding *train test split* karena memanfaatkan lebih banyak data dan bias yang ada pada dataset dapat diminimalisir [31].



Gambar 3.4. Contoh *Confusion Matrix*

Confusion Matrix menujukkan bagaimana model dapat memprediksi label sesuai dengan kenyataannya. *Confusion Matrix* divisualisasikan dengan warna dimana semakin terang (kuning) pada kasus ini maka jumlah dari nilai yang ditunjukkan semakin banyak. Sedangkan warna yang gelap menunjukkan nilai yang sedikit. Nilai dari *Confusion Matrix* kemudian dapat digunakan untuk mengukur *accuracy*, *precision*, *recall* dan *f1-score*.

Berikut adalah contoh perhitungan *accuracy*, *precision*, *recall* dan *f1-score* untuk kelas positif. Untuk menghitung pada kelas netral dan kelas negatif menggunakan cara yang sama.

$$Accuracy = \frac{13+2+76}{120} = 0.75 \tag{3.1}$$

$$Precision(positive) = \frac{76}{6+3+76} = 0.89 \tag{3.2}$$

$$Recall(positive) = \frac{76}{8+3+76} = 0.87 \tag{3.3}$$

$$F1score = 2 * \frac{0.89 * 0.87}{0.89+0.87} = 0.88 \tag{3.4}$$