BAB I

PENDAHULUAN

1.1 Latar Belakang

Di era digital yang kompetitif, perusahaan aksesoris gadget menghadapi tantangan besar dalam mengoptimalkan manajemen persediaan dan prediksi penjualan. Perencanaan stok yang akurat menjadi kunci untuk memastikan kelancaran operasional bisnis, menghindari kerugian akibat kelebihan atau kekurangan stok, serta meningkatkan daya saing perusahaan. Namun, kompleksitas pasar dan perubahan tren konsumen yang cepat membuat peramalan penjualan yang hanya berdasarkan data penjualan historis seringkali kurang efektif.

Pasar *Smartphone* Indonesia, 5 Perusahaan Teratas dalam Hal Pengiriman, Pangsa Pasar, dan Pertumbuhan YoY, Q4 2021 (pengiriman dalam jutaan)

Vendor	Pengiriman 2021	Pangsa Pasar 2021	Pengiriman 2020	Pangsa Pasar 2020	Pertumbuhan YoY
1. OPPO	8.5	20.8%	8.2	22.3%	3.5%
2. Xiaomi	8.1	19.8%	6.0	16.3%	34.2%
3. vivo	7.4	18.1%	9.3	25.2%	-20.2%
4. Samsung	7.2	17.6%	6.0	16.4%	19.5%
5. realme	5.0	12.2%	5.2	14.0%	-3.3%
Lain-Lain	4.7	11.5%	2.1	5.8%	119.0%
Total	40.9	100.0%	36.9	100.0%	10.9%

Sumber: IDC Quarterly Mobile Phone Tracker, 2021Q4

Catatan:
* Angka-angka diatas adalah hasil pembulatan

Gambar 1 Pasar *Smartphone* Indonesia

Berdasarkan laporan *International Data Corporation* (IDC), terdapat perubahan tren minat *smartphone* yang terjadi pada tahun 2021. Oppo berhasil menjadi nomor satu di tahun 2021 setelah digeser Vivo di tahun sebelumnya dengan pertumbuhan YoY sebesar 3%. Posisi kedua diraih oleh Xiaomi yang berhasil meningkatkan pertumbuhan YoY sebesar 34,2%. Vivo yang menempati posisi ketiga mengalami penurunan YoY sebesar -20,2%. Diikuti dengan Samsung dengan YoY sebesar 19,5%. Dengan berubahnya minat pasar terhadap brand smartphone yang dalam waktu setahun ini, penjualan aksesoris yang berkaitan erat dengan penjualan brand smartphone ini terancam mengalami penurunan siklus hidup. Oleh karena itu diperlukan pendekatan yang lebih holistik dengan memanfaatkan teknologi kecerdasan buatan dan analisis data, yaitu dengan memanfaatkan data historis penjualan dan opini yang tersebar di media sosial untuk melakukan prediksi penjualan agar penjualan menjadi lebih optimal.

Prediksi merupakan proses meramalkan kondisi di masa depan dengan melakukan pengujian terhadap kondisi di masa lampau. Prediksi penjualan bertujuan untuk memperkirakan besaran penjualan serta dapat digunakan dalam pengambilan Keputusan dan kebijakan berdasarkan hasil prediksi yang diperoleh (Kurniawati dkk.,

2023). Prediksi penjualan memberikan informasi mengenai bagaimana Perusahaan sebaiknya mengelola tim penjualan, produk, dan anggarannya. Prediksi yang akurat dapat meningkatkan pendapatan sesuai dengan pertumbuhan pasar pada tingkat pendapatan maksimum yang memungkinkan (Wisesa dkk., 2020).

PT. Bintang Internasional Cabang Veteran Selatan Kota Makassar merupakan salah satu cabang perusahaan penyedia berbagai variasi aksesoris gadget, spare part, dan juga *lifestyle* yang berkualitas dengan mengutamakan produk yang "Berkualitas, Harga Terjangkau Serta Pelayanan Yang Terbaik". Dalam beberapa tahun terakhir, PT. Bintang Internasional menunjukkan pertumbuhan pesat yang dapat dilihat dengan pembukaan cabang yang terjadi beberapa tahun terakhir seperti pembukaan toko di Surabaya dan Balikpapan pada tahun 2024 dan pembukaan toko di bali (toko kedua di Bali) dan Palu pada tahun 2023. Saat ini, PT. Bintang memiliki 28 cabang toko yang tersebar di Indonesia mulai dari Sulawesi, Kalimantan, hingga Jawa. PT. Bintang Aktif dalam melakukan promo penjualan di berbagai media sosial seperti TikTok dan Instagram untuk meningkatkan penjualan. Prediksi bertujuan untuk mempermudah bagian penyedia stok barang pada PT. Bintang Internasional Cabang Veteran Selatan Kota Makassar dalam melakukan perencanaan penyediaan stok barang serta memberitahu pihak perusahaan tentang produk-produk paling banyak dibeli oleh konsumen.

Media sosial merupakan salah satu dampak dari perkembangan teknologi informasi yang sangat pesat. Media sosial khususnya X menjadi platform untuk mengutarakan pendapat dan membagikan informasi (Evhen & Maharani, 2021). Penelitian yang dilakukan oleh Zhaojing Lin mengenai prediksi penjualan dengan memanfaatkan sentimen untuk menunjukkan hubungan antara penjualan dan sentimen menunjukkan bahwa adanya penurunan nilai *error* MSE dibandingkan dengan model yang tidak memanfaatkan sentimen. Oleh sebab itu, penambahan fitur sentimen terhadap sistem prediksi penjualan dipercaya dapat meningkatkan akurasi sistem karena adanya hubungan antara sentimen media sosial terhadap performa penjualan.

*Long Short-Term Memory (*LSTM*)* adalah suatu model yang merupakan pengembangan dari model *Recurrent Neural Network (*RNN*)* dirancang untuk meningkatkan algoritma RNN yang mampu memecahkan masalah gradien yang hilang dengan menambahkan *cell state* untuk mengingat atau melupakan data (Kurniawati dkk., 2023). Penelitian yang dilakukan oleh Robertus Bagaskara Radite Putra dan Hendry dalam prediksi penjualan barang retail menunjukkan bahwa model LSTM lebih unggul dibandingkan model GRU dengan MAE 12,536, MSE 316,826, dan MAPE 31,572. Penelitian yang dilakukan oleh Pooja Mehta, Sharnil Pandya, dan Ketan Kotecha mengenai prediksi stok harga saham dengan membandingkan beberapa algoritma menunjukkan bahwa LSTM menghasilkan akurasi lebih tinggi dibandingkan dengan model yang lain, yaitu sebesar 92,42%. LSTM telah menunjukkan keunggulan dalam memprediksi pola *time series* yang kompleks, termasuk dalam konteks prediksi penjualan. Oleh sebab itu, model LSTM akan digunakan dalam prediksi penjualan.

XGBoost merupakan salah satu metode *boosting* yaitu kumpulan *decision tree* yang pembangunan pohon berikutnya akan bergantung pada pohon sebelumnya. Pohon pertama dalam XGboost akan lemah dalam melakukan klasifikasi dengan inisialisasi *probability* yang ditentukan oleh peneliti dan kemudian akan dilakukan *update* bobot

pada setiap pohon yang dibangun sehingga menghasilkan kumpulan pohon klasifikasi yang kuat (Syukron dkk., 2020). Penelitian mengenai klasifikasi sentimen ulasan produk lokal yang dilakukan oleh Ivan Rifky Hendrawan menggunakan beberapa algoritma *machine learning* yaitu *Naïve bayes*, SVM, dan XGBoost menunjukkan bahwa algoritma XGBoost lebih unggul dibandingkan algoritma *machine learning* yang lain. Oleh sebab itu, algoritma XGBoost akan digunakan dalam melakukan analisis sentimen karena dapat memberikan wawasan tentang preferensi dan opini konsumen terkini. Dengan demikian, integrasi pendekatan LSTM dan XGBoost berpotensi menghasilkan prediksi penjualan yang lebih akurat dan responsif terhadap dinamika pasar aksesoris gadget yang cepat berubah.

Berdasarkan uraian di atas, sebagian besar penelitian belum banyak memanfaatkan sentimen dalam mengeksplorasi penjualan ritel. Penelitian ini akan mengeksplorasi penggunaan sentimen dalam kasus penjualan ritel. Oleh sebab itu, Penulis mengangkat judul "Prediksi Penjualan Aksesoris Gadget Menggunakan Model *Long-Short Term Memory* Dengan Integrasi Pengaruh Sentimen Media Sosial Berbasis *Extreme Gradient Boosting*". Penelitian ini bertujuan untuk membangun sebuah sistem untuk melakukan prediksi penjualan dengan memanfaatkan sentimen media sosial. Dengan adanya sistem ini, diharapkan dapat membantu perusahaan dalam mengambil keputusan.

1.2 Teori

1.2.1 PT. Bintang Internasional

Toko Bintang adalah perusahaan ritel penyedia teknologi seperti berbagai variasi aksesoris gadget, spare part dan juga *lifestyle* berkualitas yang telah memiliki puluhan toko di seluruh Indonesia. Toko Bintang berawal dari sebuah toko aksesoris gadget di sebuah mall. Karena inovasi dari pendirinya, Andri Gunawan Horas, Toko Bintang tumbuh menjadi perusahaan nasional yang memimpin bisnis retail aksesoris gadget di Indonesia. Untuk memperluas jaringan bisnisnya, setiap tahun Toko Bintang membuka toko baru di seluruh daerah di Indonesia. Tidak hanya itu, Toko Bintang juga memperluas jaringan pasar online dengan memiliki toko online di beberapa *marketplace*. Saat ini, PT. Bintang Internasional telah memiliki 28 cabang yang tersebar luas di Indonesia dengan 11 cabang di antaranya berada di sulawesi selatan.

1.2.2 Youtube

Youtube adalah sebuah situs berbagi video yang populer dimana para pengguna dapat memuat, menonton, menyukai video, mengomentari dan berbagi klip video secara gratis. *Youtube* yang merupakan salah satu layanan dari Google ini memberi fasilitas kepada penggunanya berupa fitur untuk mengupload video dan bisa diakses oleh pengguna yang lain dari seluruh dunia secara gratis (Afdhal dkk., 2022). Salah satu fitur yang diberikan oleh Youtube untuk penggunanya adalah fitur komentar, di mana pengguna bisa mengomentari sebuah klip video yang mereka buka dengan syarat pengguna harus *login* terlebih dahulu (Rahman dkk., 2020).

1.2.3 X

X (dahulu dikenal dengan twitter) adalah salah satu alat komunikasi yang digunakan untuk menyebarkan informasi kepada banyak orang tanpa terbatas ruang dan waktu. Platform ini berfungsi sebagai jejaring sosial dan mikroblogging daring yang memungkinkan pengguna untuk mengirim dan membaca pesan berbasis teks dengan batasan 280 karakter, yang disebut kicauan atau *tweet* (Rahmawati dkk., 2024).

1.2.4 Term Frequency Inverse Document Frequency

TF-IDF adalah sebuah model statistik untuk mengevaluasi seberapa penting kata dalam kumpulan dokumen. Algoritma ini terdiri dari dua bagian: (1) *Term Frequency* (TF) yang merepresentasikan frekuensi kemunculan sebuah kata dalam teks; (2) *Inverse Document Frequency* (IDF) mengukur seberapa penting sebuah kata yang diimbangi dengan seberapa sering kemunculan sebuah kata dalam keseluruhan data. Nilai signifikansi dari sebuah kata meningkat bersamaan dengan frekuensi kemunculan pada sebuah teks, namun berbanding terbalik dengan meningkatnya frekuensi kemunculan dalam kumpulan teks (Huda dkk., 2022). TF-IDF umumnya digunakan dalam penambangan teks dan pencarian informasi untuk mengevaluasi pentingnya istilah linguistik (umumnya unigram atau bigram) dalam korpus yang diteliti (Musyarofah dkk., 2020). Perhitungan TF-IDF adalah sebagai berikut.

$$tf_{t,d} = \frac{f_{t,d}}{n_d} \quad (1)$$

$$idf_t = log\,\frac{N}{df_t} \quad (2)$$

$$tf - idf_{t,d} = tf_{t,d} \,.\, idf_{t,d} \quad (3)$$

Dimana:

$f_{t,d}$ = frekuensi term dalam dokumen d

df_t = frekuensi dokumen term t, yaitu jumlah dimana dokumen term t muncul

1.2.5 Analisis Sentimen

Analisis sentimen merupakan suatu metode dalam pemrosesan bahasa alami (*natural language processing*) yang bertujuan untuk mengevaluasi dan mengidentifikasi nada emosional atau suasana hati yang terkandung dalam data teks. Dengan menganalisis kata dan frasa, metode ini mengklasifikasikan teks ke dalam kategori sentimen positif, negatif, atau netral. Pentingnya analisis sentimen terletak pada kemampuannya dalam memperoleh wawasan berharga dari data teks dalam jumlah besar, yang memungkinkan perusahaan untuk memahami sentimen pelanggan, mengambil keputusan yang tepat, serta meningkatkan kualitas produk dan layanan mereka (Jim dkk., 2024). Analisis sentimen dapat membantu untuk memperoleh gambaran umum presepsi masyarakat dengan cara mengelompokkan jenis opini menjadi kategori positif, negatif, atau netral (Darwis dkk., 2021). Dalam perkembangan analisis sentimen Indonesia, Indonesia mengalami kemajuan dengan adanya *IndoBERT* yang mampu mencapai tingkat akurasi yang lebih tinggi dalam mengklasifikasikan sentimen tweet berbahasa Indonesia dibandingkan metode *machine learning* konvensional. Model ini menggunakan arsitektur *transformer* yang memungkinkannya memahami konteks kata secara dua arah (*bidirectional*), yakni dengan mempertimbangkan kata-kata sebelum dan sesudah dalam

sebuah kalimat. Hal ini menjadi keunggulan dibandingkan model-model sebelumnya yang umumnya hanya memproses teks secara satu arah (Manoppo dkk., 2025).

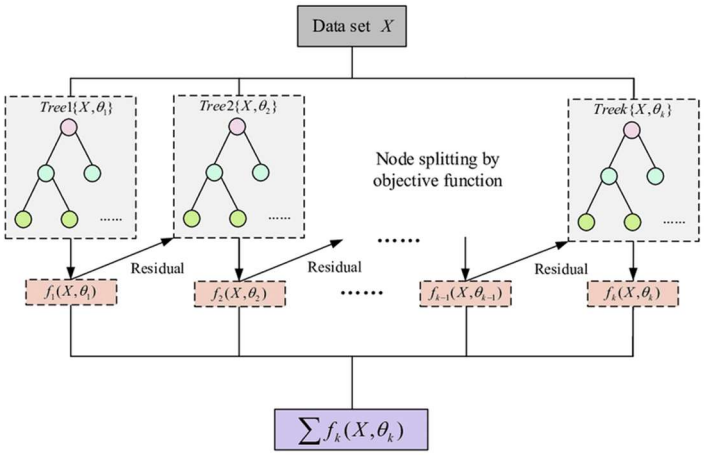
1.2.6 Sales Forecasting

Forecasting merupakan salah satu metode yang digunakan untuk memprediksi atau meramalkan suatu hal yang belum pernah terjadi. Prediksi merupakan suatu kegiatan meramalkan secara terstruktur mengenai sesuatu yang mungkin terjadi dimasa akan datang yang bersumber pada informasi dari masa lalu dan sekarang yang dimiliki, agar kesalahannya dapat diperkecil (Dewi dkk., 2022). *Sales forecasting* memainkan peran penting dalam operasional bisnis karena memengaruhi keputusan terkait manajemen persediaan, alokasi sumber daya, dan perencanaan keuangan. Prediksi penjualan yang akurat sangat penting untuk mengoptimalkan pengelolaan arus kas, menyesuaikan strategi pemasaran dan penjualan, serta mendukung perencanaan strategis (Sun & Li, 2024). Prediksi penjualan yang hanya menggunakan data penjualan historis terkadang masih belum optimal karena kurangnya faktor-faktor luar lainnya. Penelitian yang telah dilakukan oleh Zhaojing menunjukkan faktor luar seperti opini pelanggan memiliki dampak terhadap penjualan yang ditunjukkan dengan berkurangnya nilai *error* pada model.

1.2.7 Extreme Gradient Boosting

XGBoost (*Extreme Gradient Boosting*) merupakan suatu algoritma kombinasi antara *boosting* dan *gradient boosting*. Model klasifikasi yang dibangun menggunakan metode *boosting* adalah model dengan cara melakukan prediksi dari *error* model sebelumnya untuk membuat model baru. Algoritma ini dinamakan sebagai *gradient boosting*, dengan penggunaan *gradient descent* dilakukan untuk memperkecil *error* pada saat pembentukan model baru (Ubaidillah dkk., 2022). XGBoost menggunakan model yang lebih teratur untuk membangun struktur pohon regresi sehingga dapat memberikan kinerja yang lebih baik dan mampu mengurangi kompleksitas model untuk menghindari *overfitting*. Hasil prediksi akhir dari XGBoost adalah penjumlahan hasil prediksi dari setiap pohon regres (Yulianti dkk., 2022).

Dalam melakukan analisis sentimen, XGBoost mampu menghasilkan performa lebih baik dibandingkan dengan algortima machine learning lainnya seperti *support vector machine* dan *naïve bayes*. Hal tersebut ditunjukkan pada penelitian yang dilakukan oleh Ivan Rifky Hendrawan menggunakan beberapa algoritma machine learning yaitu *Naïve bayes*, SVM, dan XGBoost dalam melakukan tugas analisis sentimen. XGBoost mampu memperoleh akurasi tertinggi sebesar 0.941 dibandingkan dengan SVM dan *Naïve bayes* yang menghasilkan akurasi masing-masing adalah 0.939 dan 0.915.



Gambar 2 *Flowchart XGBoost (Guo dkk., 2020)*

XGBoost memperbaiki kelemahan algoritma *gradient boosting* tradisional dengan memperkenalkan beberapa inovasi, termasuk regularisasi, pohon keputusan berurutan, dan penanganan yang lebih baik terhadap data yang hilang (*missing data*). XGBoost juga mampu mengoptimalkan fungsi tujuan secara efisien dengan menggunakan pendekatan gradien stokastik. Hal ini memungkinkan XGBoost untuk menghasilkan model yang lebih cepat dan lebih baik dalam hal akurasi prediksi, bahkan untuk *dataset* yang sangat besar dan kompleks (Wicaksono dkk., 2024). Metode XGBoost memerlukan fungsi objektif yang berguna untuk menilai seberapa bagus model yang didapatkan sesuai dengan data latih. Karakteristik yang terpenting dari fungsi objektif terdiri dari 2 bagian yaitu nilai *loss function* dan nilai regularisasi. Berikut adalah persamaannya (Yulianti dkk., 2022).

$$obj(\theta) = L(\theta) + \Omega(\theta) \quad (4)$$

Dimana:

L = *Loss function*,
Ω = Nilai regulasi,
θ = Parameter model.

Loss function secara umum ditulis sebagai berikut.

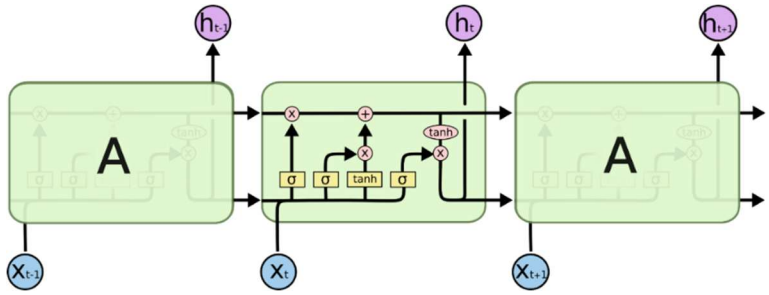
$$L(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) \quad (5)$$

Dimana:

y_i = Nilai data sebenarnya yang dianggap benar,
$\hat{y}_l$ = Hasil nilai prediksi dari model,
n = Jumlah iterasi nilai dari model.

1.2.8 Long-Short Term Memory

Long Short-Term Memory (LSTM) merupakan pengembangan dari *Recurrent Neural Network* (RNN). LSTM dirancang untuk meningkatkan algoritma *RNN* yang mampu memecahkan masalah gradien yang hilang dengan menambahkan *cell state* untuk mengingat atau melupakan data. *Cell state* memuat struktur yang disebut *cell gate*. Setiap *gate* memiliki empat bagian, yaitu *input gate*, *forget gate*, *memory-cell state gate*, dan *output gate* (Kurniawati dkk., 2023).



Gambar 3 Arsitektur *LSTM (Ku dkk., 2023)*

a) *Forget Gate*

Forget gate memiliki fungsi untuk menentukan data yang akan dihilangkan dan menentukan waktu latensi yang optimal untuk input selanjutnya. Berikut adalah fungsi aktivasi sigmoid.

$f_t = \sigma\ (W_f\ .\ [h_{t-1}, x_t] + b_t)$ (6)

Dimana:

f_t = *Forget gate*,
σ = Fungsi aktivasi *sigmoid*,
W_f = Bobot *forget gate*,
h_{t-1} = Nilai *hidden state cell* sebelum,
x_t = Nilai *input*,
b_t = Bias *forget gate*.

b) *Input Gate*

Input gate memiliki fungsi untuk membawa titik masuk data dari luar dan mengolah data yang masuk. Berikut adalah dua fungsi aktivasi.

$i_t = \sigma\ (W_i\ .\ [h_{t-1}, x_t] + b_i)$ (7)

$c_t = tanh(W_c\ .\ [h_{t-1}, x_t] + b_c)$ (8)

Dimana:

i_t = *Input gate*,
W_i = Bobot *input gate*,
b_i = Bias *input gate*,
$\tilde{C}_t$ = *Candidate gate*,
tanh = Fungsi aktivasi *tanh*,
W_c = Bobot *candidate gate*,
b_c = Bias *candidate gate*.

c) *Cell State*

Cell state memiliki fungsi untuk menggantikan nilai sebelumnya pada *memory cell* dengan nilai *memory cell* yang baru. Berikut adalah fungsi pada *cell state*.

$c_t = (i_t * \tilde{C}_t + f_t * c_{t-1})$ (9)

Dimana:

c_t = *Cell gate*,
i_t = *Input gate*,
$\tilde{C}_t$ = *Candidate gate*,

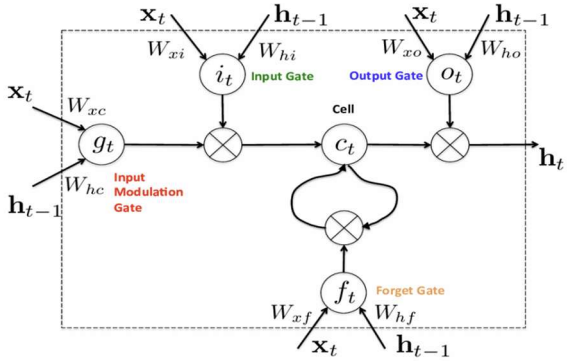
f_t = *Forget gate*,
c_{t-1} = Nilai *cell state* sebelum.

d) *Output Gate*
Output gate untuk melakukan seluruh proses *input* dan *output* perhitungan atau *output* dalam sel LSTM. Berikut adalah fungsi pada *output gate*.

$o_t = \sigma (W_o . [h_{t-1}, x_t] + b_o)$ (10)

$h_t = o_t * tanh(c_t)$ (11)

Dimana:
o_t = *Output gate*,
W_o = Bobot *output gate*,
b_o = Bias *output gate*,
h_t = *Hidden state*,
c_t = *Cell gate*.



Gambar 4 Ilustrasi Alur Arsitektur LSTM (Khumaidi dkk., 2020)

1.2.9 *Differencing*

Differencing merupakan metode yang dilakukan untuk menstasionerkan data yang tidak stasioner dalam rata-rata. *Differencing* adalah selisih antara data ke-t dengan data ke-t-1 (Nurman dkk., 2022). Berikut adalah persamaan *differencing* orde pertama.

$$\Delta Z_t = Z_t - Z_{t-1} \quad (12)$$

Dimana:
Z_t = nilai pada waktu t,
Z_{t-1} = nilai pada waktu t-1.

1.2.10 ADF

Uji ADF merupakan pengembangan versi pengujian *Dickey Fuller*. Uji ADF adalah salah satu metode yang digunakan untuk memastikan bahwa data sudah stasioner. Uji ADF bertujuan untuk mengetahui apakah data mengandung *unit roots* atau tidak. Jika data mengandung *unit roots* maka data yang digunakan belum stasioner. Apabila hasil uji ADF menunjukkan belum stationer, maka harus dilakukan proses differencing. Berikut adalah persamaan ADF (Nurman dkk., 2022).

$$\Delta Z_t = \beta_1 + \beta_2 T + \gamma Z_{t-1} + \sum_{i=2}^{p} \alpha_i \Delta 2 Z_{t-1+1} + a_t \quad (13)$$

Dimana:

β_1 = Konstanta (*Intercept*),
$\beta_2 t$ = Tren deterministik (Opsional),
γZ_{t-1} = Parameter utama untuk uji *unit root*,
$\sum_{i=2}^{p} \alpha_i \Delta 2 Z_{t-1+1}$ = Tambahan lag untuk mengatasi autokorelasi,
a_t = Nilai error pada waktu t.

1.2.9 Cross Correlation Function

Cross Correlation Function (CCF) atau korelasi silang adalah statistik yang digunakan untuk mengukur kebergantungan antara dua deret waktu pada lag waktu yang berbeda. Tujuannya adalah untuk memahami bagaimana satu deret waktu memengaruhi deret waktu lainnya atau apakah ada keterkaitan waktu antara keduanya. Berikut adalah persamaan CCF.

$$Corr_{XY}(k) = \frac{Cov(X_{t+k}, Y_t)}{\sqrt{Var(X_{t+k})}\sqrt{Var(Y_t)}} \quad (14)$$

Dengan $Cov(X_{t+k}, Y_t)$ adalah kovarians antara X_{t+k} dan Y_t. Untuk menilai signifikansi korelasi pada data aktual maupun data hasil estimasi, ditetapkan batas signifikansi yang dihitung sebagai $\pm \frac{1,96}{\sqrt{N}}$. Nilai N merupakan banyak observasil dalam data dan 1,96 adalah nilai kuantil distribusi normal standar pada tingkat kepercayaan 95% (Nurhayati dkk., 2025).

1.2.9 Confusion Matrix

Confusion Matrix adalah tabel yang membandingkan hasil prediksi model dengan data asli, memberika informasi tentang jumlah prediksi yang benar (*True Positive* dan *True Negative*) serta kesalahan prediksi (*False Positive* dan *False Negative*) (Reynaldi Valerian dkk., 2025). Matriks ini membandingkan nilai aktual dari data dengan nilai prediksi yang dihasilkan oleh model. Manfaat dari *matrix* ini yaitu mengevaluasi akurasi model, mengidentifikasi kesalahan yang dibuat oleh model, membandingkan kinerja model yang berbeda. Matriks ini mudah dipahami dan dapat memberikan informasi yang berharga tentang bagaimana model klasifikasi bekerja (Normawati & Prayogi, 2021).

Tabel 1 Confusion Matrix

		Predicted	
		Positif	**Negatif**
Actual	**Positif**	TP	FN
	Negatif	FP	TN

True Positive (TP) adalah jumlah data yang diklasifikasikan sebagai positif oleh model dan juga benar sebagai positif. *False Positive* (FP) adalah jumlah data yang diklasifikasikan sebagai positif oleh model namun salah sebagai positif. *True Negative* (TN) adalah jumlah data yang diklasifikasikan sebagai negatif oleh model dan juga benar sebagai negatif. *False Negative* (FN) adalah jumlah data yang diklasifikasikan sebagai negatif oleh model namun salah sebagai negatif.
Confusion Matrix dapat digunakan untuk menghitung nilai akurasi, presisi, *recall* dan *f1-score* dari tiap-tiap kelas.

1. Akurasi
   Akurasi menggambarkan tingkat keakuratan sistem dalam melakukan proses klasifikasi secara benar. Akurasi dapat dihitung dengan persamaan berikut.
   $$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (15)$$
2. Presisi
   Presisi menggambarkan perbandingan atau rasio antara jumlah prediksi benar bernilai positif dibandingkan dengan keseluruhan hasil yang diprediksi bernilai positif. Nilai presisi dapat dihitung dengan persamaan berikut.
   $$Presisi = \frac{Tp}{TP+ FP} \quad (16)$$
3. Recall
   Recall menggambarkan seberapa banyak prediksi yang bernilai benar dari semua kelas yang bernilai positif. Nilai recall dapat dihitung dengan persamaan berikut.
   $$Recall = \frac{TP}{TP+ FN} \quad (17)$$
4. F1 Score
   F1 Score merupakan perbandingan antara nilai rata-rata presisi dan nilai *recall*. *F1 Score* dapat dinyatakan dengan persamaan berikut.
   $$F1\ Score = \frac{2\ X\ Presisi\ X\ Recall}{Presisi+Rec} \quad (18)$$

1.2.10 *Root Mean Squared Error* (RMSE)

RMSE adalah aturan penilaian kuadrat yang juga mengukur ukuran rata-rata kesalahan. RMSE adalah akar kuadrat dari perbedaan kuadrat rata-rata antara prediksi data dan pengamatan aktual (Milniadi & Adiwijaya, 2023). Berikut adalah persamaan RMSE.

$$RMSE = \sqrt{\frac{1}{n}\Sigma(yi - \hat{y}i)^2} \quad (19)$$

Dimana:
RMSE = nilai *root mean square error*,
yi = nilai hasil observasi,
ŷi = nilai hasil prediksi,
n = jumlah data.

1.2.11 *Mean Absolute Error* (MAE)

Mean Absolute Error adalah rata-rata dari selisih absolut antara nilai prediksi dan nilai aktual. Metrik ini memberikan gambaran seberapa besar rata-rata kesalahan prediksi dari model (Shodiq dkk., 2024). Berikut adalah persamaan MAE.

$$MAE = \frac{1}{n}\Sigma|yi - \hat{y}i| \quad (20)$$

Dimana:
yi = nilai hasil observasi,
ŷi = nilai hasil prediksi,
n = jumlah data.

1.2.12 *Symmetric Mean Absolute Percentage Error* (SMAPE)

Symmetric Mean Absolute Percentage Error (SMAPE) merupakan salah satu metode yang dapat digunakan untuk mengukur ketepatan peramalan. Metode SMAPE merupakan pembaharuan dari metode MAPE. Metode SMAPE dapat mengatasi permasalahan dalam metode MAPE. Metode SMAPE dapat mengatasi permasalahan nilai hasil perhitungan metode MAPE yang tidak terdefinisi jika data aktual sama dengan 0 ($Yt = 0$). Metode SMAPE juga bisa mengatasi permasalahan metode MAPE mengenai besarnya *error*. Permasalahan metode MAPE mengenai besarnya *error* terjadi ketika nilai dari Yt (data aktual) mendekati nol (Astuti dkk., 2023). Berikut adalah persamaan *SMAPE*.

$$SMAPE = \left(\frac{2}{n}\right) \sum_{t=1}^{n} \frac{|F_t - Y_t|}{|Y_t| + |F_t|} \ (100\%) \quad (21)$$

Dimana:

Yt = data aktual periode t,
Ft = nilai peramalan pada periode t,
n = jumlah periode.

1.2.13 *Mean Absolute Scaled Error* (MASE)

Mean Absolute Scaled Error (MASE) adalah salah satu metrik yang digunakan untuk mengukur keakuratan model peramalan. MASE dihitung dengan menghitung *Mean Absolute Error* (MAE) dari model peramalan yang digunakan, kemudian membandingkannya dengan MAE dari model peramalan acak. Semakin kecil nilai MASE, semakin akurat model peramalan (Wijaya, 2023). MASE digunakan sebagai alternatif dari MAPE karena kelemahan MAPE yaitu tidak bisa digunakan saat data aktual bernilai nol atau sangat kecil, sensitif terhadap *outlier*, dan cenderung memberi bobot lebih besar pada nilai yang tinggi termasuk data yang memiliki variansi yang besar. Hal ini menyebabkan hasil MAPE akan memiliki nilai yang tinggi. Nilai MASE yang lebih dari 1 menunjukkan model kurang baik, nilai MASE sama dengan 1 maka model setara dengan *naïve forecast*, dan nilai MASE kurang dari 1 maka model lebih baik dari *naïve forecast* (Ashri dkk., 2025). Berikut adalah persamaan MASE.

MASE = *mean*($|q_t|$) (22)

Dengan:

$$q_t = \frac{e_t}{\frac{1}{n-1} \sum_{i=2}^{n} |Y_i - Y_{i-1}|} \quad (23)$$

Dimana:

e_t = $Y_t - F_t$; *Error* antara nilai aktual dan prediksi,
q_t = Skala kesalahan waktu ke-t,
n = jumlah total observasi.

1.2.14 Reoder Point (ROP)

Reorder point adalah ketersediaan barang yang harus tetap ada saat pemesanan dilakukan atau disebut dengan titik pemesanan kembali (Maulidi & Listianti, 2023). Berikut adalah persamaan ROP.

Reorder Point = (D x LT) + SS (24)

Dengan:

SS = Z x σ_d x $\sqrt{LT}$ (25)

Dimana:

D = rata-rata permintaan per periode waktu,
LT = lama waktu untuk pengiriman atau *lead time* dari pemasok,
SS = *safety stock,*
Z = Nilai tingkat layanan,
σ_d = Standar deviasi penjualan harian.

1.3 Rumusan Masalah

1. Bagaimana cara mengembangkan model prediksi penjualan aksesoris gadget yang menggabungkan data historis penjualan internal perusahaan dengan analisis sentimen media sosial?
2. Bagaimana pengaruh sentimen media sosial terhadap akurasi prediksi penjualan aksesoris gadget dengan model yang hanya menggunakan data historis penjualan?
3. Bagaimana perbandingan performa antara model LSTM dan model *hybrid* LSTM-XGBoost dalam memprediksi penjualan aksesoris gadget?

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk:

1. Untuk mengetahui cara mengembangkan model prediksi penjualan aksesoris gadget yang menggabungkan data historis penjualan internal Perusahaan dengan analisis sentimen media sosial.
2. Untuk mengetahui pengaruh sentimen media sosial terhadap akurasi prediksi penjualan aksesoris gadget dengan model yang hanya menggunakan data historis penjualan.
3. Untuk mengetahui perbandingan performa antara model LSTM dan model *hybrid* LSTM-XGBoost dalam memprediksi penjualan aksesoris gadget.

1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

1. **Bagi PT. Bintang Internasional Cabang Veteran Selatan Kota Makassar**: sebagai referensi untuk perusahaan dalam melakukan managemen inventori sehingga tidak mengalami *overstocking / understocking*.
2. **Bagi peneliti dan akademisi**: sebagai kontribusi pada pengetahuan dalam bidang *data mining* terkhususnya dalam penerapan analisis sentimen dalam prediksi penjualan.

1.6 Ruang Lingkup

Penelitian ini memiliki beberapa batasan, yaitu:

1. Penelitian ini hanya menggunakan data penjualan aksesoris gadget berupa casing hp untuk merek hp keluaran tahun 2020 sampai dengan 2021 dari PT. Bintang Internasional cabang veteran selatan kota Makassar dengan data yang digunakan adalah data dengan periode penjualan selama satu tahun, yaitu tahun 2021.
2. Ulasan yang digunakan bersumber dari media sosial X dan Youtube berbahasa Indonesia dengan sentimen yang terbagi atas 3 kategori, yaitu positif, negatif, dan netral.
3. Metode yang digunakan dalam penelitian adalah metode *Long-Short Term Memory* dan *Extreme Gradient Boosting*.

BAB II

METODE PENELITIAN

2.1 Lokasi Penelitian

Penelitian ini dilakukan sejak bulan November 2024 di Laboratorium *Ubiquitous Computing and Networking* Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

2.2 Instrumen Penelitian

Instrumen yang digunakan dalam penelitian ini adalah sebagai berikut:

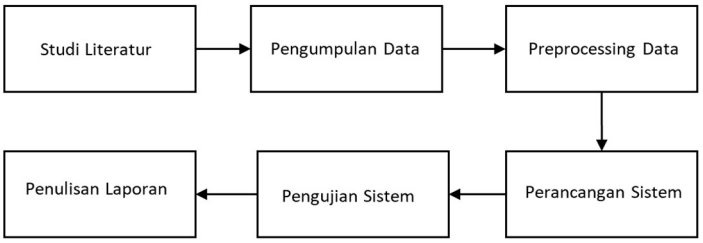
Tabel 2 Spesifikasi Software

Komponen	Spesifikasi
OS	*Windows 10*
Bahasa	*Python*
Tools	*Google Collaboratory*, *Microsoft Excel*, *Google Drive*, *Browser*, *Visual Paradigm*

Tabel 3 Spesifikasi Hardware

Komponen	Spesifikasi
Jenis	*HP Spectre x360*
Processor	*IntelCore i7*
RAM	8GB

2.3 Tahapan Penelitian



Gambar 5 Tahapan Penelitian

2.3.1 Studi literatur

Langkah pertama dalam melakukan penelitian ini adalah melakukan studi literatur mengenai penelitian terkait dari berbagai sumber seperti jurnal, *e-book*, dan artikel ilmiah. Penelitian tersebut berupa prediksi penjualan, analisis sentimen, dan algoritma yang digunakan, yaitu LSTM dan XGBoost. Dengan dilakukan pengkajian ini, diharapkan dapat memperoleh pemahaman dan referensi yang dapat membantu dalam melakukan penelitian.

2.3.2 Pengumpulan Data

Pada tahap ini dilakukan proses pengumpulan data berupa data historis penjualan dan ulasan pada media sosial terkait dengan merek hp. Pengumpulan data historis penjualan akan dilakukan pada PT. Bintang Internasional cabang Veteran Selatan dengan periode penjualan tahun 2021. Pemilihan dataset aksesoris hp hanya akan mencakup casing hp. Hal ini dikarenakan penjualan casing hp yang sensitif terhadap tren hp. Dengan munculnya hp baru, casing hp memiliki potensi untuk tidak terjual. Oleh sebab itu, penelitian ini hanya akan berfokus pada aksesoris hp berupa casing hp.

Pengumpulan ulasan pada media sosial dilakukan pada media sosial X dan *Youtube*. Proses *data scraping* akan menggunakan *Tweet Harvest* untuk X dan Youtube *Data API* untuk Youtube. Ulasan dicari dengan menggunakan kata kunci mengenai merek hp.

2.3.3 Preprocessing Data

Tahap selanjutnya adalah *preprocessing data*. *Data* yang telah diperoleh akan dilakukan pengolahan dan pembersihan sehingga data yang digunakan bersih dan siap untuk di analisis.Tahapan yang dilakukan dalam preprocessing data historis penjualan adalah *feature engineering* hari dan sentimen, *scaling*, dan *sequencing*. Proses *preprocessing data* ulasan meliputi *case folding*, *text cleaning*, *tokenization*, *stopword removal*, normalisasi, dan *stemming*. Data yang telah dipreprocessing diharapkan dapat mempermudah model dalam melakukan pembelajaran sehingga model menjadi lebih efektif dan efisien.

2.3.4 Perancangan Sistem

Tahap berikutnya adalah perancangan sistem. Data yang telah diolah akan dijadikan sebagai data latih dan data uji untuk sistem. Perancangan sistem akan dilakukan sesuai dengan rancangan sistem yang telah dibuat.

2.3.5 Pengujian Sistem

Pada tahap ini, sistem yang telah dibuat akan dilakukan proses evaluasi terhadap performa sistem untuk mengetahui seberapa bagus kinerja sistem berjalan sesuai dengan yang diharapkan. Pengujian model analisis sentimen akan dievaluasi menggunakan *confusion matrix* sedangkan pengujian model prediksi penjualan akan dilakukan dengan menghitung nilai *error* menggunakan metrik evaluasi RMSE, MAE, SMAPE, dan MASE.

2.3.6 Penulisan Laporan

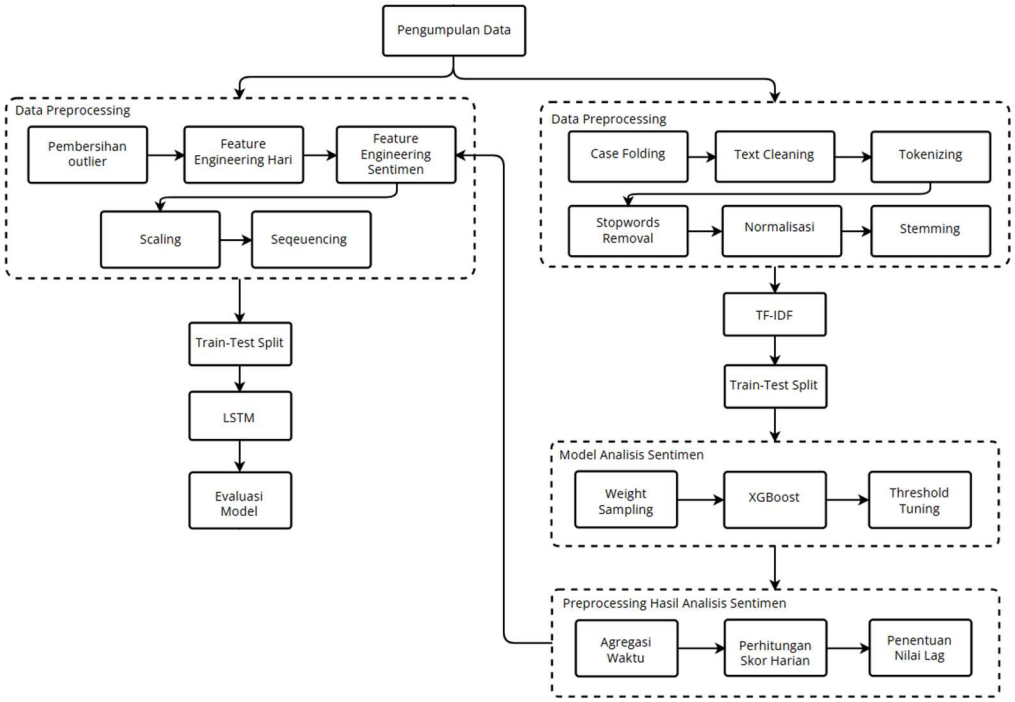
Penyusunan laporan penelitian dilakukan berdasarkan penelitian yang telah dilakukan pada tahap sebelumnya mulai dari awal hingga akhir dalam bentuk skripsi sebagai bahan publikasi.

2.4 Teknik Pengumpulan Data

Teknik pengumpulan data dilakukan dengan menggunakan *library* dan *API* yang telah tersedia. Proses *data scrapping X* akan dilakukan dengan menggunakan suatu *library python* bernama *Tweet Harvest* sedangkan proses *data scrapping* Youtube akan dilakukan dengan menggunakan *API* yang telah disediakan oleh Youtube, yaitu Youtube *Data API v3*. Proses *data scrapping* dilakukan menggunakan kata kunci nama merek hp untuk X dan video mengenai merek hp untuk Youtube.

2.5 Perancangan Sistem

Rancangan sistem dapat dilihat pada diagram alur berikut:



Gambar 6 Rancangan Sistem

Gambar 6 menunjukkan rancangan sistem penelitian. Penelitian ini akan membangun dua model, yaitu model analisis sentimen dan model prediksi penjualan. Hasil dari model analisis sentimen berupa sentimen yang telah dilabel akan dijadikan input untuk model prediksi penjualan. Penelitian ini akan melihat seberapa besar pengaruh sentimen terhadap model prediksi penjualan.

2.6 Pengumpulan Data

Proses pengumpulan data dibagi menjadi dua, yaitu pengumpulan data historis penjualan dan data ulasan media sosial.

2.6.1 Data Historis Penjualan

Pengumpulan data penjualan historis penjualan dilakukan pada PT. Bintang Internasional dengan periode waktu tahun 2021. Penelitian ini akan menggunakan tiga brand hp berupa Samsung, Redmi, dan Oppo. Data penjualan aksesories gadget yang akan diolah berupa casing hp. Berikut adalah beberapa contoh data penjualan yang telah diperoleh.

	A	B	C	D	E	F	G	H
1	index	tgl	kodeproduk	jumlah	hjual	hmodal	nama	satuan
2	1	02/01/2021	122L39HTS	1	291000	0	LCD SONY XPERIA Z1/L39H ORI PABRIK H	PCS
3	2	02/01/2021	0501TRSV8	1	11000	5000	CBL DT BT TRIANGLE SRS V8 1M	PCS
4	3	02/01/2021	0311NMTRA	1	30000	15000	HF NEW MUTIARA (*)	PCS
5	4	02/01/2021	0501KPSTPC	1	11000	5250	CBL DT BT K-POP SRS TYPE-C(*)	PCS
6	5	02/01/2021	0101G530H	1	21000	0	CS OC FULLSET SAM G530H/PRIME H/P/G	PCS
7	6	02/01/2021	127R5PAB	1	236000	235000	LCD XM REDMI 5 PLUS ORI H/P	PCS
8	7	02/01/2021	0501TRSV8	1	11000	5000	CBL DT BT TRIANGLE SRS V8 1M	PCS
9	8	02/01/2021	0501CMV82M	1	29000	13750	CBL DT BT CROSS METAL V8 2M	PCS
10	9	02/01/2021	0311MX97	1	15000	7500	HF MUSIC X97	PCS
11	10	02/01/2021	0501A1TC	1	10000	5250	CBL DT GALACTIC MDL A1/TYPE-C	PCS
12	11	02/01/2021	127OPF7	1	330000	435000	LCD OPPO F7 ORI PABRIK H	PCS

Gambar 7 Data Historis Penjualan

Data yang telah diperoleh berjumlah 558.623 baris. Data tersebut kemudian dilakukan pengelompokkan berdasarkan jenis barang berdasarkan dua angka di depan pada kode produk, yaitu 06 dan 09 untuk casing. Setelah dikumpulkan barang yang berupa casing, dilakukan pengelompokkan berdasarkan merek yang digunakan untuk casing tersebut dengan melihat kata kunci pada nama barang seperti SAM untuk Samsung, REDMI untuk Redmi, dan OP untuk Oppo. Pengemlompokkan data memperoleh total penjualan untuk masing-masing merek sebanyak 21613 aksesoris untuk Samsung, 12750 aksesoris untuk redmi, dan 15601 aksesoris untuk Oppo. Data tersebut kemudian dikelompokkan berdasarkan hari terjualnya sehingga jumlah baris yang digunakan adalah 365 baris data yang terdiri dari kolom tanggal dan total penjualan untuk periode waktu satu tahun. Berikut adalah beberapa contoh hasil dari pengolahan data tersebut.

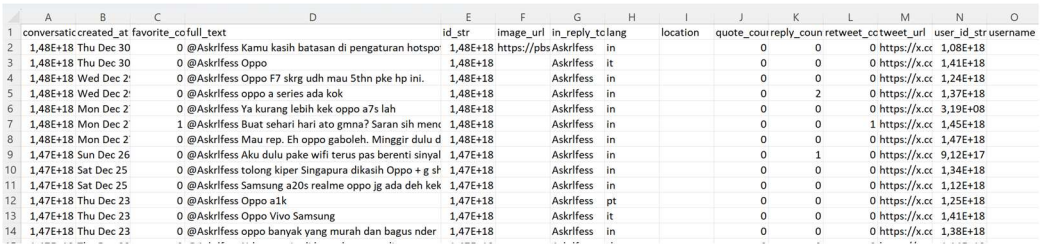
	A	B
1	**tgl**	**sales**
2	2021-01-01 00:00:00	0
3	2021-01-02 00:00:00	18
4	2021-01-03 00:00:00	4
5	2021-01-04 00:00:00	14
6	2021-01-05 00:00:00	37
7	2021-01-06 00:00:00	9
8	2021-01-07 00:00:00	4
9	2021-01-08 00:00:00	22

Gambar 8 Data Historis Setelah Diolah

2.6.2 Data ulasan Media Sosial

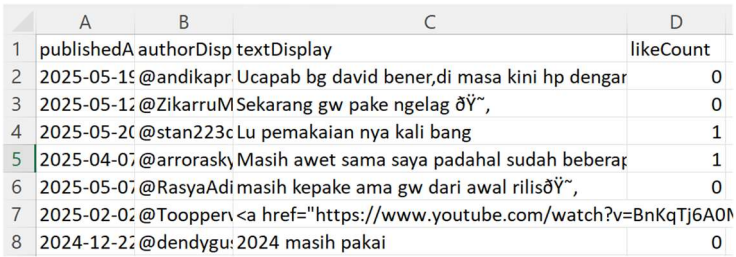
Pengumpulan data ulasan media sosial dilakukan dengan menggunakan *library* dan *API* yang telah tersedia, yaitu *Tweet Harvest* dan Youtube *Data API v3*. Proses *data scrapping X* dilakukan dengan menggunakan *library Tweet Harvest* menggunakan keyword brand hp sedangkan proses *data scrapping* Youtube dilakukan dengan menggunakan *API* yang telah disediakan oleh Youtube, yaitu Youtube *Data API v3*. Data

yang digunakan mencakup periode selama satu tahun dari 1 Januari 2021 sampai dengan 31 Desember 2021 sesuai dengan rentang waktu pada data penjualan historis. Berikut adalah beberapa contoh hasil dari proses *scrapping* yang telah dilakukan.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	conversatic	created_at	favorite_co	full_text	id_str	image_url	in_reply_tc	lang	location	quote_cou	reply_coun	retweet_cc	tweet_url	user_id_str	username
2	1,48E+18	Thu Dec 30	0	@Askrlfess Kamu kasih batasan di pengaturan hotspo	1,48E+18	https://pbs	Askrlfess	in		0	0	0	https://x.cc	1,08E+18	
3	1,48E+18	Thu Dec 30	0	@Askrlfess Oppo	1,48E+18		Askrlfess	it		0	0	0	https://x.cc	1,41E+18	
4	1,48E+18	Wed Dec 2	0	@Askrlfess Oppo F7 skrg udh mau 5thn pke hp ini.	1,48E+18		Askrlfess	in		0	0	0	https://x.cc	1,24E+18	
5	1,48E+18	Wed Dec 2	0	@Askrlfess oppo a series ada kok	1,48E+18		Askrlfess	in		0	2	0	https://x.cc	1,37E+18	
6	1,48E+18	Mon Dec 2	0	@Askrlfess Ya kurang lebih kek oppo a7s lah	1,48E+18		Askrlfess	in		0	0	0	https://x.cc	3,19E+08	
7	1,48E+18	Mon Dec 2	1	@Askrlfess Buat sehari hari ato gmna? Saran sih menc	1,48E+18		Askrlfess	in		0	0	1	https://x.cc	1,45E+18	
8	1,48E+18	Mon Dec 2	0	@Askrlfess Mau rep. Eh oppo gaboleh. Minggir dulu d	1,48E+18		Askrlfess	in		0	0	0	https://x.cc	1,47E+18	
9	1,47E+18	Sun Dec 26	0	@Askrlfess Aku dulu pake wifi terus pas berenti sinyal	1,47E+18		Askrlfess	in		0	1	0	https://x.cc	9,12E+17	
10	1,47E+18	Sat Dec 25	0	@Askrlfess tolong kiper Singapura dikasih Oppo + g sh	1,47E+18		Askrlfess	in		0	0	0	https://x.cc	1,34E+18	
11	1,47E+18	Sat Dec 25	0	@Askrlfess Samsung a20s realme oppo jg ada deh kek	1,47E+18		Askrlfess	in		0	0	0	https://x.cc	1,12E+18	
12	1,47E+18	Thu Dec 23	0	@Askrlfess Oppo a1k	1,47E+18		Askrlfess	pt		0	0	0	https://x.cc	1,25E+18	
13	1,47E+18	Thu Dec 23	0	@Askrlfess Oppo Vivo Samsung	1,47E+18		Askrlfess	it		0	0	0	https://x.cc	1,41E+18	
14	1,47E+18	Thu Dec 23	0	@Askrlfess oppo banyak yang murah dan bagus nder	1,47E+18		Askrlfess	in		0	0	0	https://x.cc	1,38E+18	

Gambar 9 Contoh Hasil *Scrape Data X*



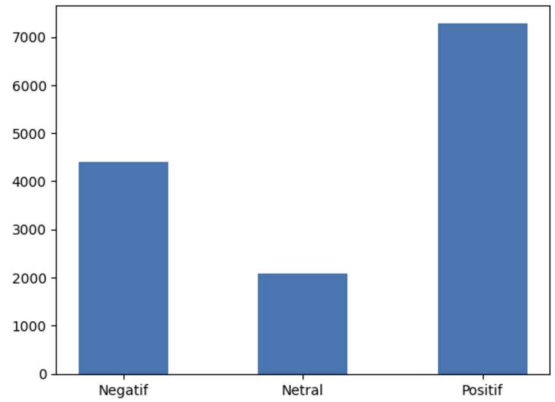
	A	B	C	D
1	publishedA	authorDisp	textDisplay	likeCount
2	2025-05-19	@andikapr	Ucapab bg david bener,di masa kini hp dengar	0
3	2025-05-12	@ZikarruM	Sekarang gw pake ngelag ðŸ˜,	0
4	2025-05-20	@stan223c	Lu pemakaian nya kali bang	1
5	2025-04-07	@arrorasky	Masih awet sama saya padahal sudah beberap	1
6	2025-05-07	@RasyaAdi	masih kepake ama gw dari awal rilisðŸ˜,	0
7	2025-02-02	@Tooppern	<a href="https://www.youtube.com/watch?v=BnKqTj6A0M	
8	2024-12-22	@dendygu	2024 masih pakai	0

Gambar 10 Contoh Hasil *Scrape Data Youtube*

Data yang berhasil dikumpulkan berjumlah 170.294 ulasan, kemudian data yang tidak mengandung sentimen dibuang sehingga menjadi 32.120 ulasan. Data yang mengandung sentimen tersebut diberi label positif, negatif, atau netral. Berikut adalah hasil dari pelabelan untuk setiap merek.

Tabel 4 Persebaran Kelas Sentimen Redmi

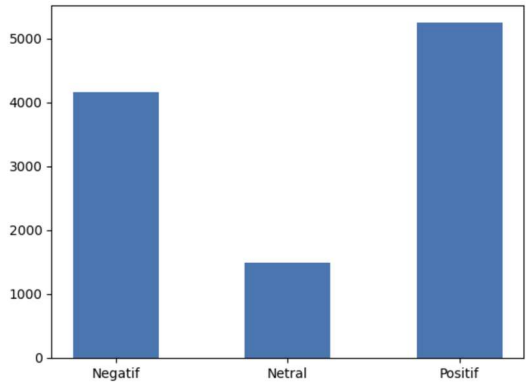
Sentimen	Jumlah
Positif	7286
Negatif	4410
Netral	2089



Gambar 11 Persebaran Kelas Sentimen Samsung

Tabel 5 Persebaran Kelas Sentimen Redmi

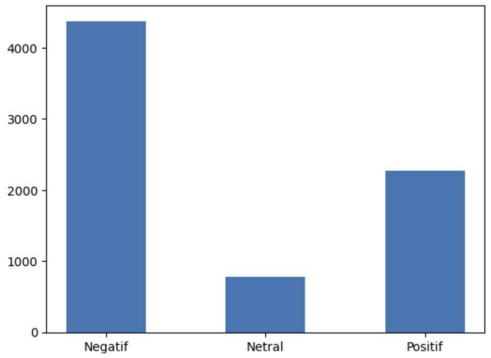
Sentimen	Jumlah
Positif	5254
Negatif	4162
Netral	1490



Gambar 12 Persebaran Kelas Sentimen Redmi

Tabel 6 Persebaran Kelas Sentimen Oppo

Sentimen	Jumlah
Positif	2268
Negatif	4377
Netral	784



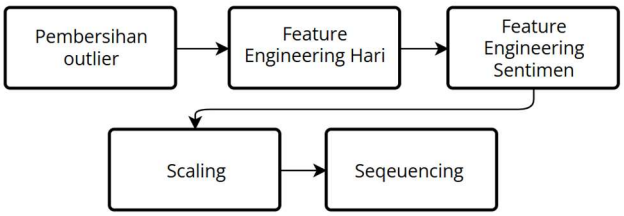
Gambar 13 Persebaran Kelas Sentimen Oppo

2.7 Preprocessing Data

Tahap *preprocessing* akan dilakukan terhadap kedua jenis data yang telah dikumpulkan pada langkah sebelumnya.

2.7.1 Preprocessing Data Historis Penjualan

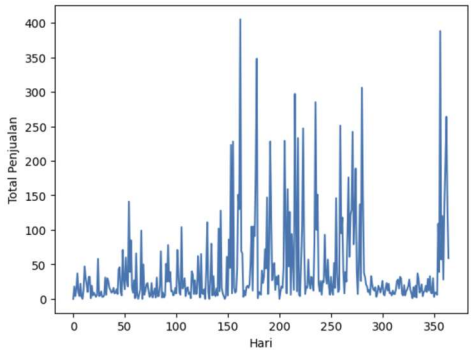
Berikut adalah langkah preprocessing data historis penjualan.



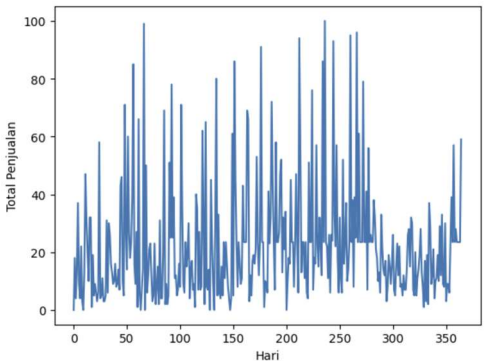
Gambar 14 Preprocessing Data Historis Penjualan

1. Pembersihan Outlier

Pembersihan *outlier* dilakukan untuk menghilangkan nilai yang dianggap ekstrem sehingga menggangu proses training model. Proses pembersihan outlier pada dataset dilakukan dengan mendeteksi nilai-nilai yang melebihi batas atas *interquartile range* (*IQR*). Nilai yang melebihi batas atas kemudian diubah menjadi nilai rata-rata. Berikut adalah hasil sebelum dan sesudah proses pembersihan *outlier*.



Gambar 15 Sebelum Pembersihan *Outlier*



Gambar 16 Sesudah Pembersihan *Outlier*

2. Feature Engineering Hari

Feature Engineering hari merupakan proses pemecahan fitur datetime menjadi komponen individual seperti hari pada satu minggu dan hari pada satu bulan. Selain itu juga, dilakukan penambahan *flag* untuk setiap hari yang dianggap memiliki pengaruh terhadap daya beli konsumen seperti hari libur, hari gajian, dan *weekend* untuk membantu model dalam memprediksi penjualan. Berikut adalah hasil *feature engineering*.

	tgl	sales	dayOfTheWeek	dayOfTheMonth	is_holiday	is_weekend	is_payday
0	2021-01-01	0.0	4	1	1	0	1
1	2021-01-02	18.0	5	2	0	1	0
2	2021-01-03	4.0	6	3	0	1	0
3	2021-01-04	14.0	0	4	0	0	0

Gambar 17 Hasil *Preprocessing* hari

3. Scaling

Scaling adalah proses mengubah nilai ke dalam suatu rentang waktu sehingga data menjadi lebih stabil. Hal ini dilakukan untuk mengurangi ketimpangan yang terjadi pada dataset untuk mempermudah model dalam proses pelatihan. Proses *scaling* dilakukan dengan menggunakan *min-max scaler*. Berikut adalah hasil dari proses *scaling* data.

```
array([[ 35.        ,  35.57142857,  15.        , ...,   0.        ,
         1.        ,   0.        ],
       [ 26.        ,  40.71428571,  16.14285714, ...,   0.        ,
         1.        ,   0.        ],
       [ 20.        ,  39.        ,  16.57142857, ...,   0.        ,
         0.        ,   0.        ],
```

Gambar 18 Data Sebelum *Scaling*

```
array([[0.08454106, 0.63085399, 0.79674797, ..., 0.        , 1.        ,
        0.        ],
       [0.06280193, 0.73002755, 0.86178862, ..., 0.        , 1.        ,
        0.        ],
       [0.04830918, 0.6969697 , 0.88617886, ..., 0.        , 0.        ,
        0.        ],
```

Gambar 19 Data Setelah *Scaling*

4. Sequencing

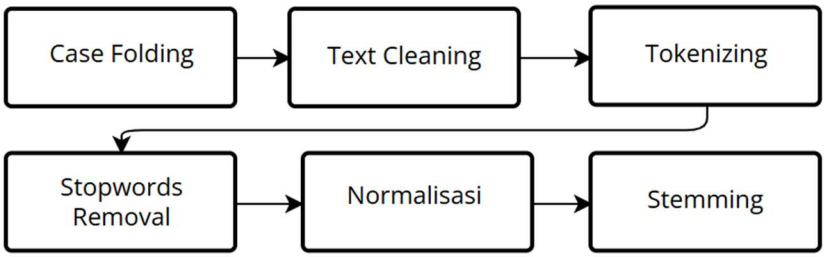
Sequencing adalah proses penyusunan data dalam rentang waktu berdasarkan urutan waktu. Hal ini dilakukan untuk membantu model dalam menangkap pola waktu. Proses *sequencing* dilakukan dengan menggunakan *timestep* 14. Berikut adalah hasil dari proses *sequencing data*.

```
[[0.04830918, 0.6969697 , 0.88617886, ..., 0.        ,
  0.        , 0.        ],
 [0.00241546, 0.71625344, 0.91869919, ..., 0.        ,
  0.        , 0.        ],
 [0.0531401 , 0.59504132, 0.74796748, ..., 0.        ,
  0.        , 0.        ],
 ...,
 [0.09178744, 0.58402204, 0.63414634, ..., 1.        ,
  0.        , 0.        ],
 [0.07729469, 0.39944904, 0.3495935 , ..., 0.        ,
  1.        , 0.        ],
 [0.08937198, 0.90082645, 0.89430894, ..., 0.        ,
  1.        , 0.        ]],
```

Gambar 20 Data Setelah *Sequencing*

2.7.2 Preprocessing Ulasan Media Sosial

Berikut adalah langkah *preprocessing* ulasan media sosial.



Gambar 21 Preprocessing Ulasan Media Sosial

1. Case Folding

Case folding merupakan proses mengubah semua huruf besar menjadi huruf kecil. Hal ini dilakukan untuk membuat *dataset* memiliki format yang sama. Berikut adalah contoh proses *case folding*.

Tabel 7 *Casefolding*

Sebelum	Sesudah
Aku udah pakek oppo a12 Suka banget	aku udah pakek oppo a12 suka banget
Tergantung pemakaian kamu	tergantung pemakaian kamu
Mahal amat	mahal amat

2. Text Cleaning

Text cleaning berfungsi untuk menghapus tanda baca pada teks beserta dengan simbol-simbol yang tidak relevan pada ulasan. Pada proses ini juga dilakukan penghapusan terhadap *tag html*, *retweet*, *mention*, dan *link* yang dianggap tidak bermakna. Hal ini dilakukan untuk mempermudah model melakukan pembelajaran. Berikut adalah contoh proses *text cleaning*.

Tabel 8 *Text Cleaning*

Sebelum	Sesudah
aku udah pakek oppo a12 suka banget	aku udah pakek oppo a suka banget
tergantung pemakaian kamu	tergantung pemakaian kamu
mahal amat	mahal amat

3. Tokenizing

Tokenizing merupakan proses dimana kalimat dipisah menjadi kata-kata yang disebut dengan *token*. Berikut adalah contoh proses *tokenizing*.

Tabel 9 *Tokenizing*

Sebelum	Sesudah
aku udah pakek oppo a suka banget	['aku', 'udah', 'pakek', 'oppo', 'a', 'suka', 'banget']
tergantung pemakaian kamu	['tergantung', 'pemakaian', 'kamu']
mahal amat	['mahal', 'amat']

4.Normalisasi

Normalisasi merupakan proses mengubah kata yang mengandung makna yang sama atau ejaan yang salah menjadi kata standar untuk membuat kata-kata tersebut menjadi satu entitas yang sama. Berikut adalah contoh proses normalisasi.

Tabel 10 Normalisasi

Sebelum	Sesudah
['aku', 'udah', 'pakek', 'oppo', 'a', 'suka', 'banget']	['aku', 'sudah', 'pakai', 'oppo', 'a', 'suka', 'sangat']
['tergantung', 'pemakaian', 'kamu']	['tergantung', 'pakai', 'kamu']
['mahal', 'amat']	['mahal', 'amat']

5. Stopwords Removal

Stopwords removal adalah proses penghapusan kata-kata yang dianggap kurang bermakna atau tidak relevan dalam dataset. Hal ini dilakukan untuk mengurangi *noise* yang terdapat pada *dataset*. Berikut adalah contoh proses *stopwords removal*.

Tabel 11 *Stopwords Removal*

Sebelum	Sesudah
['aku', 'sudah', 'pakai', 'oppo', 'a', 'suka', 'sangat']	['pakai', 'oppo', 'suka']
['tergantung', 'pakai', 'kamu']	['tergantung', 'pakai']
['mahal', 'amat']	['mahal']

6. Stemming

Stemming adalah proses mengubah kata yang memiliki imbuhan menjadi bentuk dasarnya. Hal ini dilakukan untuk menggelompokkan kata yang memiliki makna yang sama menjadi satu entitas. Berikut adalah contoh proses *stemming*.

Tabel 12 *Stemming*

Sebelum	Sesudah
['pakai', 'oppo', 'suka']	pakai oppo suka
['tergantung', 'pakai']	gantung pakai
['mahal']	mahal

2.8 TF-IDF

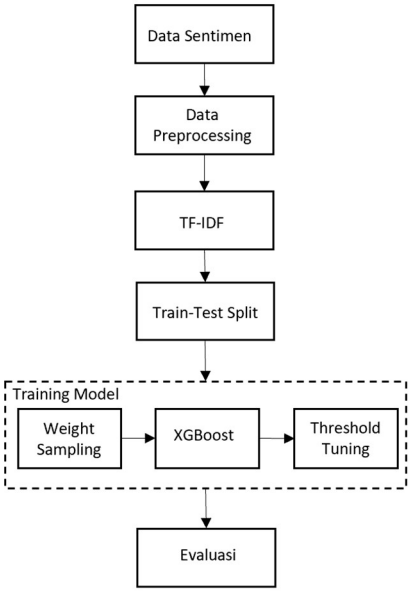
Setelah melakukan *preprocessing*, data ulasan akan dilakukan representasi ke dalam bentuk numerik dengan menggunakan teknik *TF-IDF*. *TF-IDF* dilakukan untuk menilai

bobot relevansi *term* dari sebuah dokumen terhadap seluruh dokumen dalam korpus agar mempermudah dalam proses pelatihan. Berikut adalah contoh hasil dari proses *TF-IDF*.

Tabel 13 TF-IDF

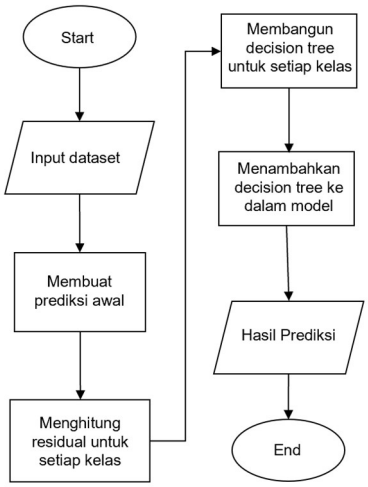
	pakai	**oppo**	**suka**	**gantung**	**mahal**
pakai oppo suka	0.474	0.622	0.622	0	0
gantung pakai	0.605	0	0	0.795	0
mahal	0	0	0	0	1

2.9 Model Analisis Sentimen dengan XGBoost



Gambar 22 Alur Analisis Sentimen dengan *XGBoost*

Pada gambar 22 menampilkan proses analisis sentimen berbasis *XGBoost*. Setelah ekstrasi fitur dengan menggunakan ektraksi fitur *TF-IDF*, langkah selanjutnya adalah melatih model *XGBoost* dengan menggunakan *library python xgboost*. Model dilatih untuk melakukan klasifikasi sentimen positif, negatif, dan netral. Untuk mengatasi ketimpangan kelas yang terjadi dalam *dataset* antara kelas netral degan kelas positif dan negatif, akan dilakukan *sampling weight* untuk mengatur bobot pada saat *training*. Setelah dilakukan proses *training*, *threshold tuning* dilakukan untuk meningkatkan nilai *recall*. Hasil dari analisis sentimen kemudian diagregasi menjadi sentimen per hari berdasarkan sentimennya untuk dipakai dalam prediksi penjualan menggunakan algoritma LSTM sebagai fitur tambahan untuk membantu dalam proses prediksi penjualan.



Gambar 23 Alur Kerja XGBoost

Gambar 23 menjelaskan proses XGBoost dalam melakukan prediksi. Pertama, XGBoost akan melakukan prediksi awal yang biasanya dilakukan dengan menggunakan nilai rata-rata. Setelah menentukan prediksi awal, XGBoost akan menghitung nilai residual untuk masing-masing *class* dengan menggunakan nilai prediksi awal tersebut. Hasil residual ini digunakan sebagai acuan dalam pembuatan *decision tree*. Setelah semua *decision tree* selesai dibuat untuk masing-masing class, *decision tree* tersebut ditambahkan ke dalam XGBoost. *Decision Tree* yang baru akan dibuat dengan mempertimbangkan nilai residual dari *decision tree* yang lama. Hal ini yang membuat *decision tree* mampu menekan nilai error dan memperbaiki akurasi pada *decision tree* sebelumnya. Proses iterasi ini akan dilakukan hingga mencapai batas jumlah *decision tree* yang telah ditentukan sebelumnya.

Berikut ini adalah contoh perhitungan sederhana untuk algoritma XGBoost. Dengan menggunakan data pada tabel 13, maka langkah pembentukan model XGBoost adalah sebagai berikut. Langkah pertama, XGBoost akan menentukan prediksi awal dengan nilai rata-rata setiap label, yaitu 0,333. Setelah menentukan prediksi awal, XGBoost akan mulai membuat decision tree baru dengan menggunakan nilai *gradient* dan *hessian*. Berikut adalah contoh perhitungan *gradient* dan *hessian* untuk kalimat “pakai oppo suka”.

Tabel 14 Contoh Gradient dan Hessian Pakai Oppo Suka

Kelas	Label One-Hot	Prediksi Awal	*Gradient* (g)	*Hessian* (h)
0	0	0.33	0.33 – 0 = 0.33	0.33 x (1 – 0.33) = 0.22
1	0	0.33	0.33 – 0 = 0.33	0.33 x (1 – 0.33) = 0.22
2	1	0.33	0.33 – 1 = -0.67	0.33 x (1 – 0.33) = 0.22

Setelah mendapatkan nilai gradient dan hessian, XGBoost akan mulai membuat *decision tree* baru berdasarkan dari nilai residual tersebut. Titik *split* terbaik akan dicari berdasarkan fitur-fitur yang digunakan, yaitu “pakai”, “oppo”, “suka”, “gantung”, dan

"mahal", dengan memaksimalkan nilai gain yang diperoleh untuk masing-masing pemisahan. Misalkan akan digunakan titik *split*, yaitu "oppo > 0,311", berikut adalah contoh perhitungan gain pada titik *split* tersebut.

$\sum g_{root}$ = -0.67 + 0.33 + 0. 33 = 0

$\sum h_{root}$ = 0.22 + 0.22 + 0.22 = 0.66

$Similarity_{Root} = \frac{(\sum g_{root})^2}{\sum h_{root} + \lambda}$

$= \frac{(0)^2}{(0.66) + 1}$

= 0

$Similarity_{Kiri} = \frac{(\sum g_l)^2}{\sum h_l + \lambda}$

$= \frac{(0.66)^2}{(0.44) + 1}$

= 0.302

$Similarity_{Kanan} = \frac{(\sum g_r)^2}{\sum h_r + \lambda}$

$= \frac{(0.67)^2}{(0.22) + 1}$

= 0.368

Gain $= \frac{1}{2}\left(Similarity_{Kiri} + Similarity_{kanan} - Similarity_{Root}\right) - \gamma$

$= \frac{1}{2}$ (0.302 + 0.368 - 0) – 0

= 0.335

Dikarenakan nilai *gain* bernilai positif, maka akan *split* diterima. Proses *splitting* akan dilakukan hingga mencapai batas nilai *max_depth*. Selanjutnya XGBoost akan menghitung *output value* untuk menentukan prediksi baru.

Output Value $= -\frac{\sum g}{\sum h + \lambda}$

$= -\frac{0.66}{0.44 + 1}$

= - 0.458

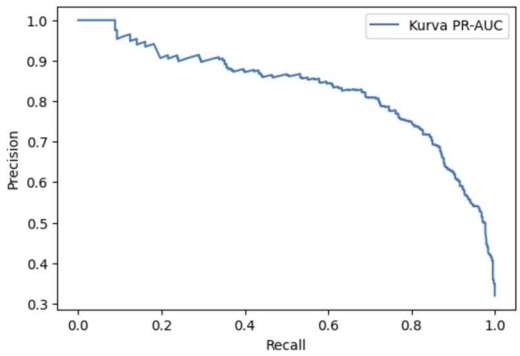
Prediksi Baru = Prediksi lama + *learning rate* x *Output Value*

= -1.099 + 0.1 x (-0.458)

= -1.145

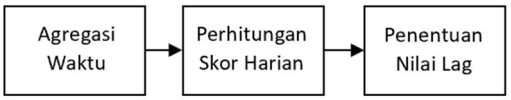
Prediksi baru yang telah dihitung kemudian akan digunakan dalam menghitung nilai *gradient* dan *hessian* pada *decision tree* yang baru. Proses ini akan diulangi hingga mencapai *n_estimators*. Setelah semua decision tree dibuat, XGBoost akan melakukan prediksi kelas dengan mengabungkan kontribusi dari setiap *decision tree* untuk setiap kelas. Model XGBoost akan memprediksi berdasarkan kelas dengan probabilitas tertinggi.

Untuk mengatasi masalah *data imbalance* yang terjadi antara kelas netral dengan kelas positif dan negatif, dilakukan *threshold tuning*. *Threshold tuning* dilakukan dengan cara menghitung nilai F1-score untuk setiap nilai *threshold*. Pemilihan nilai *threshold* dilakukan dengan memilih nilai F1-Score yang paling tinggi. Berikut adalah contoh kurva PR-AUC yang digunakan dalam penentuan nilai threshold.



Gambar 24 Grafik PR-AUC

2.10 Integrasi Sentimen Pada Model Prediksi Penjualan



Gambar 25 Integrasi Sentimen Pada Model Prediksi Penjualan

Pada gambar 25 menampilkan proses integrasi hasil analisis sentimen *XGBoost* dengan algoritma LSTM dalam melakukan prediksi penjualan. Hasil analisis sentimen *XGBoost* berupa sentimen positif, negatif, dan netral harian akan dilakukan agregasi menjadi data sentimen harian. Proses agregasi waktu dilakukan dengan cara menjumlahkan total penjualan yang ada dalam suatu waktu. Hal ini dilakukan agar sentimen memiliki interval waktu yang sama dengan data historis penjualan sehingga sentimen dapat disatukan dengan data historis penjualan berdasarkan dengan tanggal, seperti misalnya sentimen pada tanggal 1 januari 2021 akan digabungkan dengan data historis penjualan pada tanggal tersebut juga.

Untuk memadukan sentimen publik ke dalam model prediksi penjualan, sentimen akan diubah menjadi nilai skor untuk setiap brand dengan menggunakan persamaan yang telah dilakukan pada penelitian sebelumnya, yaitu sebagai berikut.

$$x_t = \frac{p_t}{p_t + n_t} \tag{23}$$

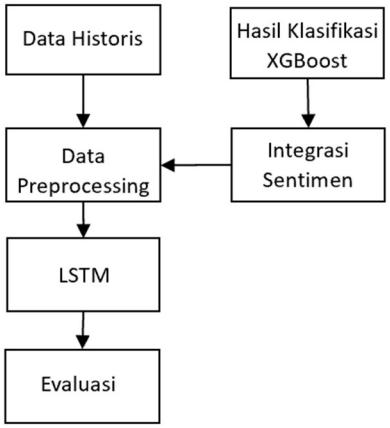
Dimana nilai x_t adalah skor sentimen, p_t adalah jumlah sentimen positif pada t, dan n_t adalah jumlah sentimen negatif pada t. Skor sentimen harian yang telah diperoleh, kemudian dilakukan pencarian nilai *lag* yang optimal dengan menggunakan metode CCF. Sebelum melakukan pencarian dengan menggunakan metode CFF, data dilakukan uji stationeritas dengan menggunakan uji ADF. Jika data menunjukkan stationeritas maka akan langsung dilakukan pencarian lag dengan metode CCF. Akan tetapi, jika data tidak menunjukkan stationeritas, maka data akan dilakukan *differencing* terlebih dahulu untuk membuat data menjadi stationer. Berikut ini adalah masing-masing *p-value* untuk hasil uji ADF untuk data penjualan dan sentimen.

Tabel 15 Hasil Uji ADF

Data	Tanpa *Differencing*	*Differencing*-1
Samsung	0,018	-
Oppo	1,876 x 10^{-22}	-
Redmi	0,013	-
Sentimen Samsung	3,848 x 10^{-30}	-
Sentimen Oppo	1,090 x 10^{-18}	-
Sentimen Redmi	0,098	3,880 x 10^{-19}

Pada Tabel 15 menunjukkan nilai *p-value* dari pengujian ADF. Hasil uji ADF menunjukkan bahwa semua data memiliki nilai p-value < 0.05 kecuali sentimen untuk redmi sehingga dapat dikatakan bahwa semua data tersebut stationer. Sentimen redmi menunjukkan stationeritas setelah dilakukan differencing pertama. Data ini kemudian akan digunakan dalam melakukan pencarian nilai lag dengan CCF.

2.11 Model Prediksi Penjualan dengan LSTM dengan Integrasi Analisis Sentimen



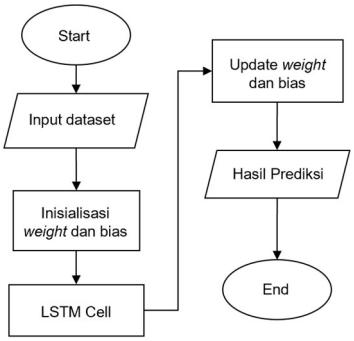
Gambar 26 Alur Prediksi Penjualan LSTM dengan Integrasi Sentimen

Gambar 26 menunjukkan proses pelatihan model LSTM dengan integrasi sentimen. Data historis penjualan akan dilakukan preprocessing dengan menambahkan skor sentimen yang telah diolah sebelumnya sehingga fitur yang digunakan dalam pelatihan adalah *sales*, *rating_lag*, *dayOfTheWeek*, *dayOfTheMonth*, *is_holiday*, *is_weekend*, dan *is_payday*. Berikut ini adalah Gambaran data yang akan digunakan dalam proses pelatihan model LSTM.

sales	rating_lag	dayOfTheWeek	dayOfTheMonth	is_holiday	is_weekend	is_payday
26.0	0.852941	4	8	0	0	0
8.0	0.493333	5	9	0	1	0
1.0	0.586207	6	10	0	1	0
26.0	0.564516	0	11	0	0	0
26.0	0.647059	1	12	0	0	0

Gambar 27 Data untuk Pelatihan Model LSTM

Setelah dilakukan *preprocessing data*, langkah selanjutnya adalah membangun model LSTM dengan menggunakan *framework deep learning* yang telah tersedia, yaitu *Tensorflow*. Model dibangun dengan menggunakan *layer* LSTM untuk memahami data time series. *Layer* LSTM dilanjutkan dengan *layer dropout* untuk mengurangi potensi model mengalami *overfitting*. Selanjutnya setiap *neuron* dihubungkan menjadi satu menggunakan *layer dense* untuk melakukan prediksi terhadap penjualan. Model yang telah dibangun dilatih dan diuji menggunakan data historis penjualan untuk melihat performa model LSTM tanpa menggunakan fitur tambahan sentimen publik.



Gambar 28 Alur Kerja LSTM

Gambar 28 menunjukkan alur kerja dari LSTM. Pertama, LSTM akan menginisialisasikan *weight* dan bias awal secara acak sebelum melakukan perhitungan. Setelah menginisialisasi *weight* dan bias, model akan mulai perhitungan untuk masing-masing gate pada LSTM. Berikut ini akan dilakukan contoh sederhana perhitungan masing-masing gate pada LSTM pada *timestep* 1 dengan nilai input 0.18705036. Diketahui nilai *weight* dan bias masing-masing gate pada inisialisasi awal adalah sebagai berikut.

Tabel 16 Contoh *Weight* dan Bias LSTM

Gate	$Weight_t$	$Weight_{t-1}$	Bias
Input Gate	-0.2957127	0.5253258	0
Forget Gate	-0.87522745	-0.1399123	1
Candidate Gate	-0.9254141	-0.41175964	0
Output Gate	-0.18638039	-0.7313763	0

Input Gate

f_t = σ (W_f . [h_{t-1}, x_t] + b_t)
= σ ((-0.2957127 x 0.18705036) + (0.5253258 x 0) + (0))
= σ (-0.0553)
= 0.4862

Forget Gate

i_t = σ (W_i . [h_{t-1}, x_t] + b_i)
= σ ((-0.87522745 x 0.18705036) + (-0.1399123 x 0) + (1))
= σ (0.8363)

= 0.6978

Candidate Gate

c_t = $tanh(W_c . [h_{t-1}, x_t] + b_c)$
= σ ((-0.9254141 x 0.18705036) + (-0.41175964 x 0) + (0))
= σ (-0.1732)
= -0.1714

Cell State

c_t = $(i_t * \tilde{C}_t + f_t * c_{t-1})$
= ((0.4862 x (-0.1714)) + (0.6978) x 0)
= -0.0833

Output Gate

o_t = $\sigma (W_o . [h_{t-1}, x_t] + b_o)$
= σ ((-0.18638039 x 0.18705036) + (-0.7313763 x 0) + (0))
= σ (-0.0349)
= 0.4913

Hidden State Baru

h_t = $o_t * tanh(c_t)$
= 0.4913 x tanh(-0.0833)
= -0.0408

Setelah mendapatkan nilai *hidden state* baru untuk *long term memory* dan *short term memory*, perhitungan berlanjut ke timestep 2 dengan menggunakan nilai *weight* dan bias yang sama hingga *batch size* habis. Pelatihan model LSTM akan berlangsung hingga mencapai *epoch* terakhir yang telah ditentukan.

Berdasarkan penelitian yang telah dilakukan sebelumnya, melakukan *tuning parameter* dapat membantu model dalam meningkatkan akurasi dalam melakukan prediksi. Penelitian yang dilakukan oleh Khumaidi melakukan *tuning* terhadap *parameter batch size*, *epoch*, jumlah *hidden layer*, dan jumlah *dropout* menemukan bahwa model memiliki performa terbaik pada *batch size* 30, *epoch* 150, 3 *hidden layers* and 3 *dropouts*. Penelitian yang dilakukan oleh Deswal dalam melakukan *hyper-parameter tuning* dengan menggunakan *epoch*, *batch size*, jumlah *neuron*, *learning rate*, dan *dropout rate* menunjukkan bahwa prediksi terbaik diperoleh dengan *epoch* 200, *batch size* 32, jumlah *neuron* 512, *learning rate* 0.002, dan *dropout rate* 0.01. Berdasarkan hal tersebut, maka akan dilakukan tuning parameter pada daerah tersebut. Pada penelitian ini, *parameter* yang akan dilakukan proses *tuning* adalah sebagai berikut.

Tabel 17 Konfigurasi *Parameter Tuning*

Parameter	Jumlah
Batch size	16, 32
Learning rate	0.001, 0.0001
Jumlah *Layer*	1, 2, 3

Proses *tuning parameter* akan dilakukan dengan menggunakan metode *grid search*, yaitu dengan cara melakukan proses *looping* untuk setiap kombinasi parameter dengan total kombinasi adalah 12 kombinasi. Metode tersebut dipilih karena komputasi yang ringan untuk jumlah *parameter* yang sedikit. Berikut ini adalah program *grid search.*

```
for batch_size in batch_size_options:
  for learning_rate in learning_rate_options:
    for num_layer in num_layer_options:
      model = Sequential()
      model.add(Input(shape=(X_train_seq.shape[1], X_train_seq.shape[2])))
      if num_layer > 1:
        for i in range(num_layer - 1):
          model.add(LSTM(units=64, activation='relu', return_sequences=True))
          model.add(Dropout(0.2))
      model.add(LSTM(units=64, activation='relu', return_sequences=False))
      model.add(Dropout(0.2))
      model.add(Dense(units=1))
      model.compile(optimizer=Adam(learning_rate=learning_rate), loss='mae')
      hist = model.fit(X_train_seq, y_train_seq, epochs=100, batch_size=batch_size,
validation_data=(X_val_seq, y_val_seq), verbose=0, callbacks=[checkpoint,
TqdmCallback(verbose=0)])
```

Pada proses training, data akan dibagi menjadi 3 bagian, yaitu data train, data test, dan data evaluasi. Data evaluasi akan mengambil 28 baris data dari belakang dan sisa data tersebut akan dibagi untuk data train dan data test dengan rasio 80:20. Dengan mengintegrasikan sentimen ke dalam model prediksi penjualan, diharapkan dapat membantu model dalam melakukan prediksi. Berikut adalah proses tuning parameter yang dilakukan.

2.12 Evaluasi Model

Evaluasi model dilakukan dengan menggunakan confusion matrix untuk mengevaluasi performa model analisis sentimen dengan XGBoost. Model prediksi penjualan LSTM dievaluasi dengan mengggunakan empat metrik evaluasi, yaitu RMSE, MAE, SMAPE, dan MASE pada prediksi 28 hari ke depan. Pada proses evaluasi model prediksi penjualan LSTM, akan dilakukan pengujian dengan membandingkan 5 kasus yang akan dicoba. Pertama, model LSTM, model hanya akan dilatih dengan menggunakan data penjualan historis saja. Kedua, model LSTM + hari, model akan dilatih dengan menggunakan data penjualan historis dan hasil *feature engineering* hari berupa hari libur, hari gajian, dan hari weekend. Ketiga, model LSTM + sentimen, model akan dilatih dengan menggunakan data penjualan historis dan hasil *feature engineering* sentimen. Keempat, model LSTM + hari & sentimen, model akan dilatih menggunakan data penjualan historis beserta dengan hasil *feature engineering* hari dan sentimen. Kelima, model akan dilakukan perbandingan cepat dengan menggunakan model *naïve last value*, yaitu model yang akan menggunakan nilai terakhir training untuk dijadikan sebagai hasil prediksi selama 28 hari ke depan, sebagai tolak ukur cepat terhadap model yang diusulkan.