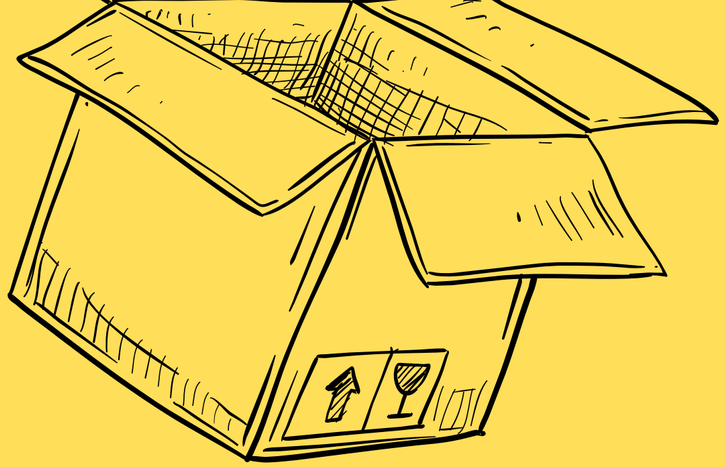
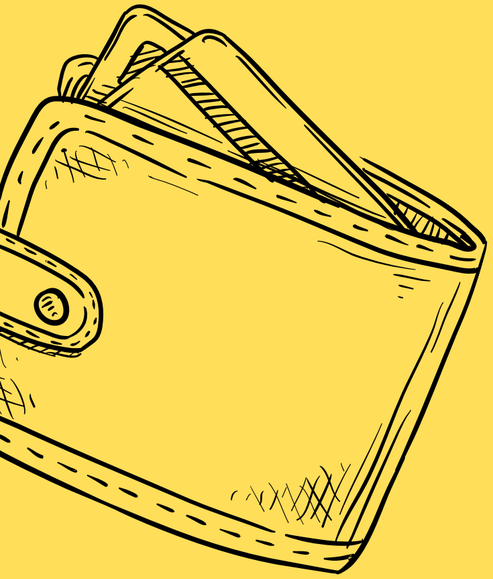


ONLINE PRODUCT PRICE COMPARISON SYSTEM



Smart Price Comparison & Prediction Platform

PRESENTED BY -

- Ayush Kumar (23/IT/045)
- Ayush Raj (23/ECE/043)
- Veerbhadra Singh (23/ECE/181)

PROBLEM STATEMENT

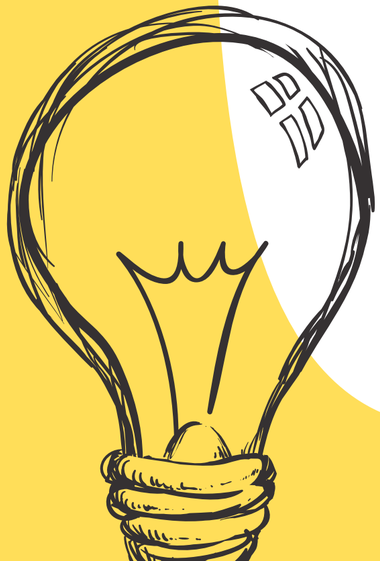
PROBLEM DEFINITION

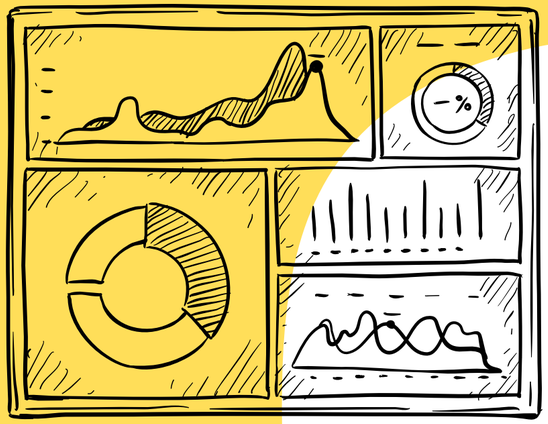
- E-commerce platforms list same products at different prices
- Hard for users to identify best deals
- Price fluctuations over time cause indecision
- Manual comparison is impractical

GOAL

Build a system that:

- Compares product prices across retailers
- Predicts price movements using machine learning





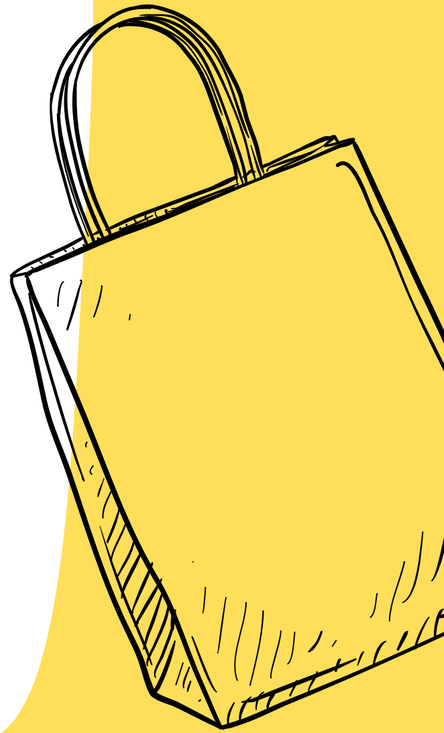
SYSTEM ARCHITECTURE

OVERVIEW

- User sends a product query
- Backend fetches data from APIs or stored datasets
- Price prediction model runs
- Outputs comparison & future price forecast

MAJOR COMPONENTS

- Data Ingestion
- ML Regression Pipeline
- API Backend
- User Interface



FLOWCHART

Data → Preprocess → Feature Engineering → Model → Prediction → Output

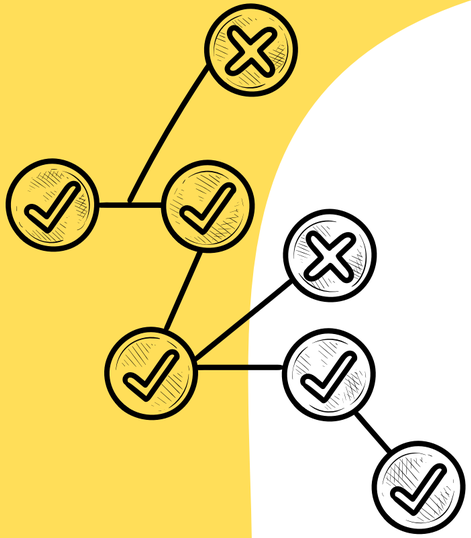
DATASET SUMMARY

Data Sources:

- Retailer price records (Amazon, Flipkart, Snapdeal, etc.)
- Historical price records

Key Fields:

Feature	Type	Description
product_name	Categorical	Name of product
retailer	Categorical	Amazon/Flipkart/etc.
current_price	Numerical	Price on given date
historical_prices	Time series	Previous prices
discount_rate	Numerical	% discount



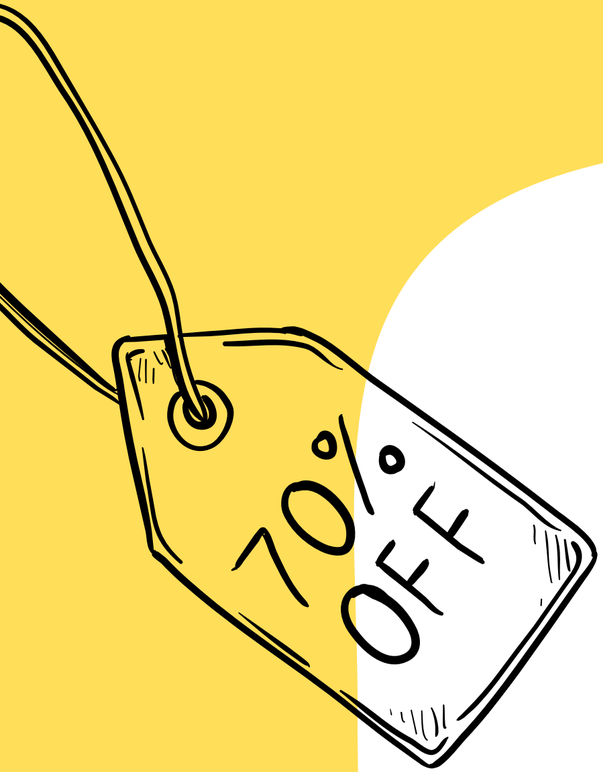
DATA PREPROCESSING

Steps Taken

- ✓ Removed duplicates
- ✓ Handled missing values
- ✓ One-Hot Encoding for categorical variables
- ✓ Standardization / Scaling
- ✓ Split data (80% train / 20% test)

Rationale

- Clean data improves model learning
- Encoding needed for non-numeric features
- Scaling ensures stable training



FEATURE ENGINEERING

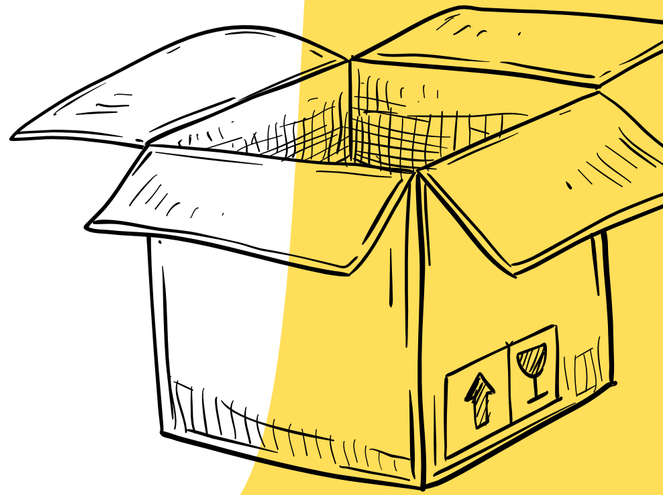
Created Features

- ✓ Price difference from average
- ✓ Rolling mean of past prices
- ✓ Discount influence
- ✓ Retailer behavior metrics

Purpose

Better represent patterns in price changes

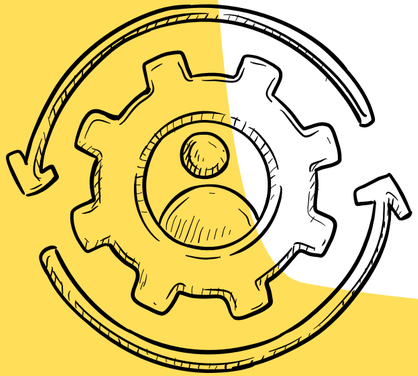
Enable models to capture non-linear trends



MACHINE LEARNING MODELS USED

Regression Algorithms Evaluated

- Linear Regression (baseline)
- Random Forest Regressor
- Gradient Boosting Regressor
- XGBoost Regressor



Why Tree-Based Models?

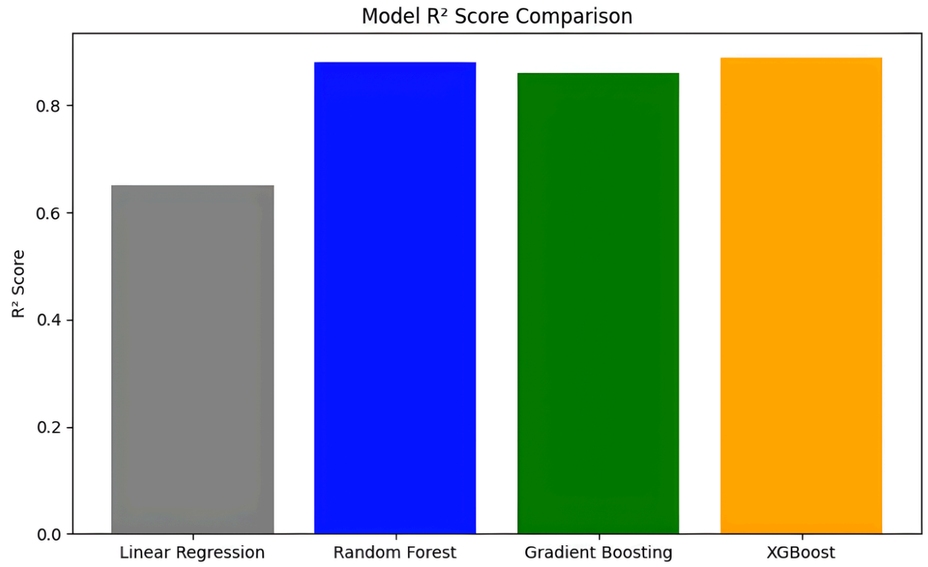
- ✓ Handles non-linear relationships
- ✓ Robust to outliers
- ✓ Better performance for complex patterns

MODEL EVALUATION (CHART)

Graph Description

Bar chart showing performance of different models:

- Linear Regression
- Random Forest
- Gradient Boosting
- XGBoost



Inference

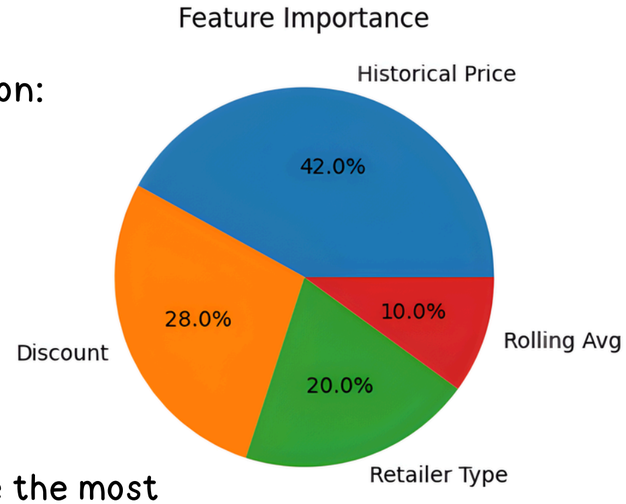
- Tree-based models outperformed linear regression
- Random Forest / Gradient Boosting most stable

FEATURE IMPORTANCE ANALYSIS

What it shows

Top features that influenced price prediction:

- Historical Price
- Discount
- Retailer Type
- Rolling Average Price



Insight

Features directly tied to price behavior are the most predictive.

PRICE TREND VISUALIZATION

Graph Type

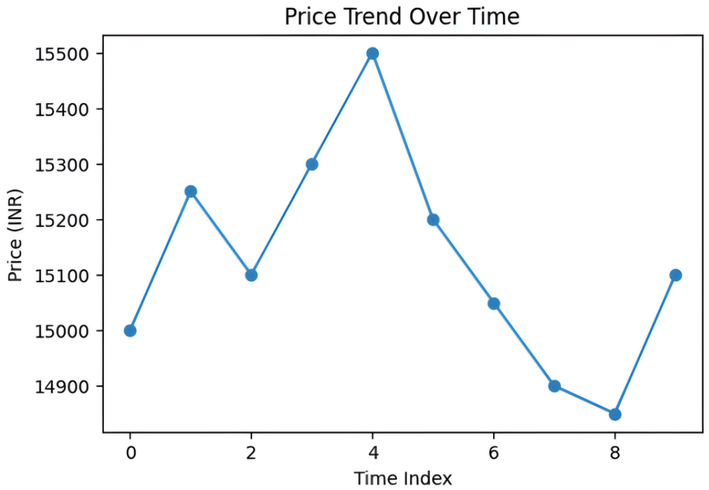
Line chart showing how price has fluctuated historically

For a product over multiple recorded dates

Explanation

Price history gives users insight into best buy periods

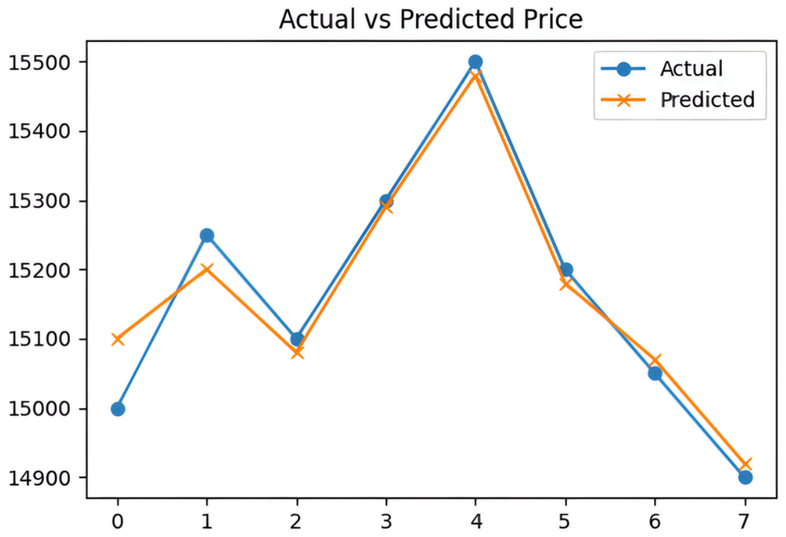
Also supports prediction via trend



PRICE PREDICTION OUTPUT

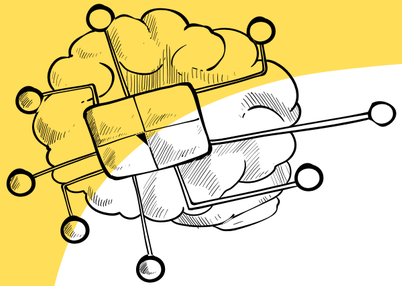
Graph Explained

- X-axis: data index/time
- Y-axis: price
- Two lines:
  - Actual price
  - Predicted price



Conclusion

The model closely tracks real price behavior.



CONCLUSION & FUTURE WORK

Achievements

- ✓ System compares prices effectively
- ✓ Predicts future price behavior with good accuracy
- ✓ Tree-based models proved suitable
- ✓ Feature engineering improved prediction quality

Future Enhancements

- Real-time web scraping
- Alerts for price drops
- Integration with mobile app