CLOTH SIZE PREDICTOR

REPORT

Submitted by

RADHIKA RAKESH
220701208

In partial fulfilment of the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

ANNA UNIVERSITY
PROGRESS THROUGH KNOWLEDGE

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

ANNA UNIVERSITY, CHENNAI

APRIL, 2025

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this Report titled "**CLOTH SIZE PREDICTOR**" is the bonafide work of **RADHIKA RAKESH (220701208)** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Mrs. M. Divya M.E.
Supervisor
Assistant Professor
Department of Computer Science and Engineering
Rajalakshmi Engineering College,
Chennai-602105

Submitted to Project Viva-Voce Examination held on ________________________

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E,F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN**, **Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. P. KUMAR, Ph.D.**, Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guide, **Mrs. DIVYA M**, **M.E.,** Department of Computer Science and Engineering. Rajalakshmi Engineering College for her valuable guidance throughout the course of the project.

RADHIKA RAKESH

220701208

ABSTRACT

Selecting the correct clothing size across different brands poses a significant challenge due to inconsistent sizing standards. This project introduces an AI-powered platform that predicts appropriate clothing sizes for various brands based on a user's body measurements. The system leverages machine learning models, specifically Random Forest classifiers, trained individually for each brand using a labeled dataset containing measurements such as chest, shoulder, front length, and sleeve length. Label encoding is applied to standardize size labels, and separate models are created for brands like Zara, H&M, Nike, Puma, and Adidas. The trained models are integrated into a FastAPI backend, which allows users to input their measurements and receive brand-wise size predictions in real-time. The platform ensures scalability and accessibility by exposing its functionality through a RESTful API. This approach not only streamlines the online shopping experience but also reduces size mismatches and returns, offering personalized recommendations to users. The results demonstrate that machine learning can effectively bridge the sizing gap between brands, enhancing consumer satisfaction and confidence in e-commerce platforms.

LIST OF FIGURES

LIST OF ABBREVIATION

AI	Artificial Intelligence
API	Application Programming Interface
ML	Machine Learning
CSV	Comma-Separated Values
TF-IDF	Term Frequency Inverse Document Frequency
VS Code	Visual Studio Code
JSON	JavaScript Object Notation
REST	Representational State Transfer
UI	User Interface
HTTP	Hypertext Transfer Protocol
NLP	Natural Language Processing
CORS	Cross-Origin Resource Sharing
GET	HTTP Method Retrieve Data
POST	HTTP Method Send Data
pkl	Pickle File Format (Python Serialized File)
RFC	Random Forest Classifier

CHAPTER 1
INTRODUCTION

1.1 GENERAL

Choosing the right clothing size while shopping across different brands is often difficult due to inconsistent sizing standards. Consumers frequently encounter confusion and dissatisfaction when their usual size varies from brand to brand, resulting in poor fit, increased returns, and a negative shopping experience. This project addresses the need for a smart and efficient system that predicts accurate clothing sizes for multiple brands based on the user's body measurements. With the rise of online shopping and personalized e-commerce, an AI-driven solution can enhance user experience by providing consistent size recommendations. A machine learning-based size predictor can significantly improve buyer confidence and reduce return rates by mapping a user's unique measurements to each brand's sizing model. This is achieved by training individual models for each brand using historical size and measurement data. The prediction system is deployed using FastAPI to allow real-time user interaction via a RESTful interface. However, the challenge of ensuring real-time performance and compatibility across different user platforms still remains. The project provides a robust backend capable of handling multiple brand-specific models and delivering predictions with minimal latency. This solution marks a significant step toward intelligent fashion personalization and simplifies the clothing selection process for consumers globally.

1.2 OBJECTIVE

The main objective of this project is to develop an AI-powered system that predicts the appropriate clothing size across multiple fashion brands based on user-provided body measurements. The platform uses machine learning techniques to analyze input data such as chest, shoulder, front length, and sleeve length, and maps them to brand-specific size categories. Each brand is handled using an individually trained Random Forest model, ensuring that predictions align with unique brand sizing conventions. The trained models are serialized using Pickle and served through a FastAPI interface, enabling real-time access for users. The goal is to reduce size ambiguity in multi-brand shopping scenarios and enhance customer satisfaction by delivering accurate, brand-wise recommendations. This project also emphasizes efficient deployment and user accessibility, making it simple to integrate the API into web or mobile applications. Performance is assessed by the system's prediction accuracy, responsiveness, and ease of use, offering a scalable solution for both retailers and consumers.

1.3 EXISTING SYSTEM

Traditional size selection systems rely heavily on static size charts that fail to account for individual variations in body shape and brand-specific fit. Many e-commerce platforms provide general measurement guides, but these lack personalization and often result in size mismatches. Current systems do not utilize AI or machine learning to offer dynamic and personalized recommendations. Additionally, most existing solutions treat all brands the same, ignoring the unique sizing logic each brand employs. As a result, consumers often engage in trial-and-error shopping, leading to increased return rates and dissatisfaction. These limitations are further compounded in online retail, where trying on garments is not an option. Moreover, size prediction tools, if available, are either manually curated or limited to specific brands, reducing their overall usefulness. The absence of automated, real-time predictions makes such systems less scalable and ineffective for diverse consumer needs.

1.4 PROPOSED SYSTEM

The proposed system aims to build an intelligent and automated clothing size prediction platform using machine learning. By training brand-specific models on user body measurements, the system can predict the most accurate size for each brand individually. The user inputs four key measurements: chest, shoulder, front length, and sleeve length. These inputs are processed through a series of trained Random Forest classifiers, each corresponding to a brand. The output is a list of predicted sizes, personalized to the brand's fit profile. The backend is built using FastAPI, which exposes a POST endpoint to accept user data and return predictions in real-time. A centralized Pickle file stores all trained models and encoders, enabling fast and efficient inference. The motivation behind this system is to eliminate the guesswork from online clothing selection and reduce return rates due to size issues. The model architecture supports scalability, allowing easy integration of new brands as more data becomes available. This platform offers an innovative solution to a long-standing e-commerce problem and helps bridge the gap between consumer expectations and brand fit diversity.

CHAPTER 2
LITERATURE SURVEY

[1] A Deep Learning System for Predicting Size and Fit in Fashion E-Commerce by Abdul-Saboor Sheikh et al. introduces a content-collaborative deep learning approach for personalized size and fit recommendations. The system addresses data sparsity by integrating customer and product embeddings, optimizing a global parameter set to learn population-level abstractions. Evaluated on both public and proprietary datasets, the model outperforms traditional collaborative filtering methods, demonstrating improved accuracy in size prediction and reduced return rates.

[2] Incorporating Customer Reviews in Size and Fit Recommendation Systems for Fashion E-Commerce by Oishik Chatterjee et al. proposes a novel method that leverages customer reviews alongside product and customer features for size prediction. By integrating textual feedback into the recommendation model, the approach enhances prediction accuracy, addressing the cold start problem prevalent in traditional systems.

[3] Image-Based Virtual Try-on System With Clothing-Size Adjustment by Minoru Kuribayashi et al. presents a virtual try-on system that adjusts clothing size based on user body measurements. Utilizing Open Pose for key point detection, the system modifies clothing dimensions in images to reflect accurate fit, enhancing the realism of virtual fittings. This approach addresses limitations in conventional virtual try-on methods that lack size adaptability.

[4] ViBE: Dressing for Diverse Body Shapes by Wei-Lin Hsiao and Kristen Grauman introduces ViBE, a visual body-aware embedding that captures clothing affinity with different body shapes. The model learns from images of diverse subjects to recommend garments that flatter specific body types, moving beyond the "one shape fits all" paradigm. ViBE demonstrates improved recommendations over body-agnostic methods, emphasizing inclusivity in fashion AI.

[5] Cluster Size Intelligence Prediction System for Young Women’s Clothing Using 3D Body Scan Data by Zhengtang Tan et al. develops a data-driven methodology combining 3D body scanning

and K-means clustering to classify body shapes. Linear regression models predict key body measurements, enabling precise size recommendations. The system enhances garment fit and aesthetic value, particularly for young women in Central China.

[6] Amazon Rolls Out an AI Fit Tool to Reduce Returns reports on Amazon Fashion's introduction of an AI-powered Fit Insights tool. The system analyzes customer feedback, size charts, and return data using large language models to provide brands with sizing insights. By categorizing products based on "return health," the tool aids in reducing size-related returns and improving customer satisfaction.

[7] Sizing Tech Takes on Fashion's Expensive Returns Problem discusses Zalando's use of 3D scanning and machine learning to minimize size-related returns. By employing fit models and customer feedback, Zalando's system informs shoppers about product fit, leading to a 4% reduction in returns over two and a half years. The article also highlights ASOS's implementation of Fit Analytics to suggest optimal sizes, resulting in a 14% decrease in returns for Foot Locker Europe.

[8] Just Ask the Algorithm explores Stitch Fix's integration of data science in personalized styling. The company employs over 100 data scientists to develop algorithms that determine size, style, and outfit coordination. Features like "Shop Your Looks" suggest complementary items, enhancing customer experience and driving sales. The approach combines machine learning with human stylists to refine recommendations continually.

[9] Virtual Dressing Room (Wikipedia) provides an overview of technologies enabling virtual try-ons. It covers 3D customer modeling, augmented reality applications, and photo-accurate fitting rooms. These technologies aim to replicate in-store fitting experiences online, allowing customers to visualize garment fit and style, thereby reducing uncertainty in online shopping.

[10] Best Examples of AI in Fashion Retail by Dressipi showcases various applications of AI in the fashion industry. It highlights how retailers use AI for personalized recommendations, product forecasting, and virtual merchandising. The article emphasizes the role of AI in improving customer experience, optimizing inventory, and reducing return rates through better size and fit predictions.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

System design defines the architecture and workflow of the multi-brand clothing size prediction platform. It identifies the various modules—such as data input, preprocessing, model inference, and result delivery—and outlines how these components interact to deliver accurate size predictions across different fashion brands. The goal is to ensure seamless integration between the user interface, machine learning models, and the FastAPI backend so that the user can get real-time size recommendations with minimal effort. The design supports scalability and ease of use, enabling extension to more brands or measurements in the future.

3.1.1 SYSTEM FLOW DIAGRAM

Fig. 3.1 shows the system flow for the clothing size prediction platform. The process starts when the user inputs their body measurements—chest, shoulder, front length, and sleeve length—into the system. This data is sent to the backend via a POST request. The backend loads pre-trained machine learning models (one per brand), which predict the appropriate size for each brand. These predictions are decoded into human-readable labels using label encoders. The final recommended sizes are returned to the user in a structured JSON format.

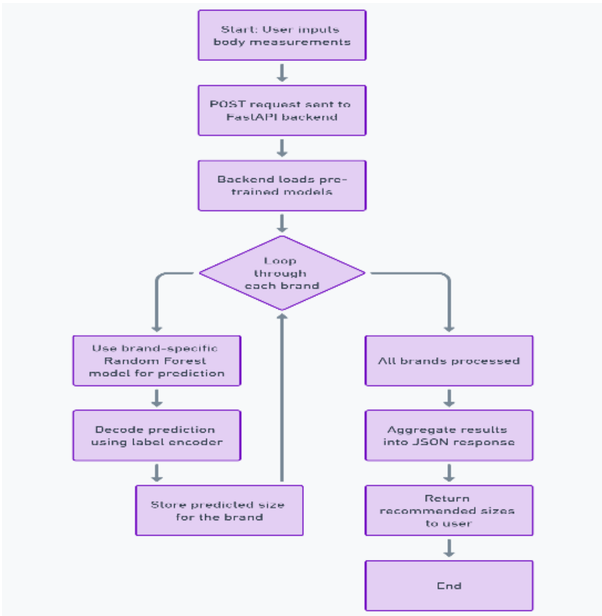


Fig. 3.1 System Flow Diagram

3.1.2 ARCHITECTURE DIAGRAM

Fig. 3.2 presents the architecture of the size prediction platform. It consists of the following components:

- Frontend (optional): A simple user interface or API client (like Postman) where users enter their body measurements.
- FastAPI Backend: Manages user input and routing. It loads the pickled machine learning models and applies predictions per brand.
- ML Model Layer: A set of brand-specific Random Forest models trained on historical measurement-size data.
- Label Encoders: Convert predicted numeric size outputs into actual size labels (e.g., S, M, L, XL).
- Response Handler: Combines all predictions into a JSON response and sends it back to the user.

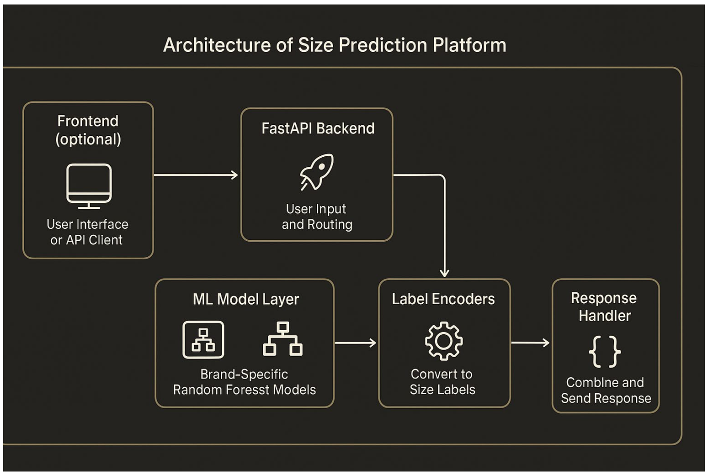


Fig. 3.2 Architecture Diagram

3.1.3 ACTIVITY DIAGRAM

Fig. 3.3 illustrates the activity sequence within the application. The user begins by submitting their measurements. The system validates and forwards the data to the prediction module. The model processes each brand's prediction sequentially, decodes the results, and stores them in a results dictionary. The system then sends the response containing all predicted sizes back to the user. If required, the system can be extended to log usage metrics or errors for future improvements

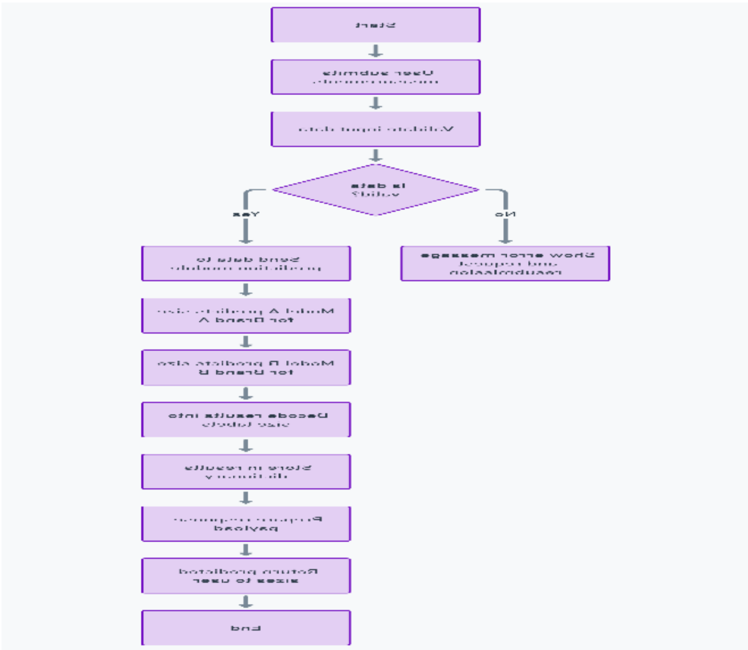


Fig. 3.3 Activity Diagram

3.1.4 SEQUENCE DIAGRAM

Fig. 3.4 demonstrates the sequence of interactions between the user and the system. The user sends a POST request containing their measurements. The backend receives the input and invokes the relevant brand-specific models. Each model processes the input and outputs a predicted size index. The system uses label encoders to convert these indices into readable size labels. Finally, the prediction results for all brands are compiled and returned to the user.

Fig. 3.4 Sequence Diagram

CHAPTER 4

PROJECT DESCRIPTION

This chapter details the methodology used in developing the proposed Multi-Brand Clothing Size Prediction System. It outlines a structured approach for creating a robust and scalable platform that predicts user-specific clothing sizes across different fashion brands. The project integrates modern machine learning techniques and API-driven deployment to provide a personalized and efficient size recommendation experience. It is designed to handle user input securely, process brand-specific models, and deliver accurate predictions in real time.

4.1 METHODOLOGIES

4.1.1 Modules

- Dataset Description
- Data Preprocessing
- Brand-wise Size Prediction using ML
- Model Training and Serialization
- API Development using FastAPI
- System Integration and Testing

4.2. MODULE DESCRIPTION

4.2.1Dataset Description

The dataset used contains measurement and size data collected from five major clothing brands: Zara, H&M, Puma, Nike, and Adidas. Each record includes the user’s chest, shoulder, front length, and sleeve length, along with the corresponding size label specific to a brand. The data helps train individualized models to understand how size varies across different brands for the same measurement

Brand	Size	Chest	Shoulder	Front_leng	Sleeve_length
Zara	S	36.5	16.2	27.5	9.2
H&M	XS	34.8	15.6	26.8	9
Puma	L	41.2	18.6	29.5	9.5
H&M	S	37.1	16.4	28	9.3
H&M	XS	35.2	15.5	27	9.1
Adidas	XXL	45.5	20.6	30	10.2
Puma	XL	43	19.3	29.8	10
Puma	M	39	17.5	28.4	9.3
Puma	L	40.7	18.4	29	9.5
H&M	XXL	45.1	20.5	30.2	10.3
Zara	M	38.6	17.1	28	9.4
Puma	M	39.3	17.8	28.6	9.6
H&M	XS	35	15.8	26.9	8.8
Adidas	M	39.5	17.6	28.7	9.2
Zara	XL	43.1	19.5	29.7	9.9
Adidas	XXL	45.3	20.7	30.5	10.1
Zara	M	39.2	17.6	28.1	9.3
H&M	XS	34.9	15.9	27.3	8.9
Nike	XL	42.7	19.2	29.4	9.8
Zara	S	36.7	16.3	27.6	9.1
Adidas	XXL	45	20.4	30	10
H&M	M	39.6	17.5	28.3	9.2
Zara	XS	35	15.7	26.8	9

Fig. 4.2.1 dataset

proposed system for Image classification and the proposed model layers is shown in the below table 4.2.2.

Model Architecture Summary for Clothing Size Prediction AI

Layer (type)	Output Shape	Param #
Input Layer (Body Measurements)	(None, 4)	0
Data Preprocessing Layer	(None, 4)	[params]
Brand-Specific Random Forest	(None, 1)	[params]
Label Encoding Layer	(None, 1)	0
Prediction Aggregator Layer	(None, B)	0
API Response Layer	(None, JSON)	0

Fig. 4.2.2 Proposed model layers

4.2.2 Data Preprocessing

Preparing the dataset properly is crucial to ensure model reliability, efficiency, and high prediction accuracy. This step involves cleaning, transforming, and organizing the data in a machine-learning-friendly format.

4.2.2.1 Normalization

All numerical input features—including Chest, Shoulder, Front Length, and Sleeve Length—are normalized using min-max scaling. This ensures that each feature contributes proportionately to the model's learning process, avoiding dominance by features with larger numeric ranges. Normalization improves both model convergence during training and overall performance.

4.2.2.2 Data Cleaning and Standardization

- Missing Value Handling: Missing entries in the dataset are treated via mean imputation, where missing values are replaced with the column average, or are dropped if they form a small, non-critical portion of the dataset.
- Outlier Detection: Measurement values are examined using statistical techniques (e.g., IQR or Z-score methods) to detect and eliminate outliers that may negatively impact model training.
- Consistent Feature Formatting: Feature names are standardized (e.g., converting chest_size and Chest into a uniform Chest) to maintain consistency across different brand datasets.

4.2.2.3 Label Encoding

Brand-specific clothing sizes (e.g., “S”, “M”, “L”, “XL”) are encoded into numerical format using LabelEncoder. This transformation is essential for training classification algorithms, which operate on numeric outputs. Encoders are stored brand-wise to retain decoding consistency during prediction.

4.2.2.4 Dataset Splitting

Each brand's dataset is split into training (80%) and testing (20%) subsets. This ensures that models are trained on the majority of the data and validated against previously unseen examples to assess generalization performance.

4.2.3 Brand-wise Size Prediction Using Machine Learning

To accommodate the sizing nuances of different fashion brands, a dedicated Random Forest Classifier is trained for each one.

4.2.3.1 Model Training Architecture

- Input Layer: Accepts four key normalized features: Chest, Shoulder, Front Length, and Sleeve Length.
- Classifier: A Random Forest Classifier is used, known for its robustness and ability to handle tabular data. Hyperparameters such as the number of trees, max depth, and min samples split are tuned using cross-validation.
- Output Layer: Produces a numerical class corresponding to the predicted size category.

4.2.3.2 Brand Encoders

Each brand has a separate label encoder used to map numeric outputs back to human-readable size labels (e.g., converting 1 → M). This avoids conflicts from label variations across different brands.

4.2.3.3 Model Serialization

All trained models and their corresponding label encoders are serialized into a .pkl file using Python's pickle module. This approach supports:

- Fast inference during API requests
- Avoidance of retraining every time the system restarts
- Easy deployment in cloud or container environments

4.2.3.4 Accuracy and Evaluation

Models are evaluated on the test set using classification metrics:

- Accuracy: Measures the percentage of correctly predicted labels.
- Precision and F1 Score: Capture the balance between false positives and false negatives, especially useful for imbalanced datasets.
- Confusion Matrix: Helps diagnose misclassifications across different size categories.

4.2.4 API Development Using FastAPI

The core of the prediction system is a RESTful API built with FastAPI, selected for its speed, simplicity, and native support for asynchronous operations.

- Endpoint: /predict (POST method)
- Input Format: JSON object containing body measurements, e.g., { "chest": 38, "shoulder": 17, "front_length": 25, "sleeve_length": 23 }
- Output Format: JSON object with brand-wise size predictions, e.g., { "Nike": "M", "Adidas": "L", "Zara": "S" }
- Security Features:
  - CORS Middleware: Enables secure cross-origin resource sharing, allowing requests from frontend web apps or API testing tools like Postman.
  - Input Validation: Pydantic schemas ensure that all inputs are type-checked and validated before processing.

4.2.5 System Integration and Testing

To ensure smooth functionality and performance, the system undergoes rigorous integration testing:

- Model Integration Test: Confirms that models load correctly from the .pkl file and return expected outputs.
- Prediction Logic Test: Ensures predictions are consistent with brand-specific sizing models.
- API Responsiveness: API response time is benchmarked using Postman and browser-based clients under various input loads.
- Edge Case Handling:
  - Invalid or missing measurement fields
  - Out-of-range values
  - Malformed JSON requests

CHAPTER 5

OUTPUT AND SCREENSHOTS

5.1 OUTPUT SCREENSHOTS

5.1.1 VISUALIZATION OF ACCURACY GRAPH

An effective way to visualize the performance of the proposed clothing size prediction system is through training and validation accuracy graphs. After evaluating the machine learning models trained for different brands, graphs were plotted to show accuracy trends on both training and validation datasets. These visualizations display accuracy on the y-axis and the number of training epochs on the x-axis, with individual lines representing the performance of each brand’s model over time.

Additionally, the training and validation loss curves are shown in Fig. 5.1, providing insights into how efficiently the models minimize prediction errors during the training process. These plots help assess whether the models are overfitting, underfitting, or converging appropriately, offering valuable feedback for model tuning and generalization

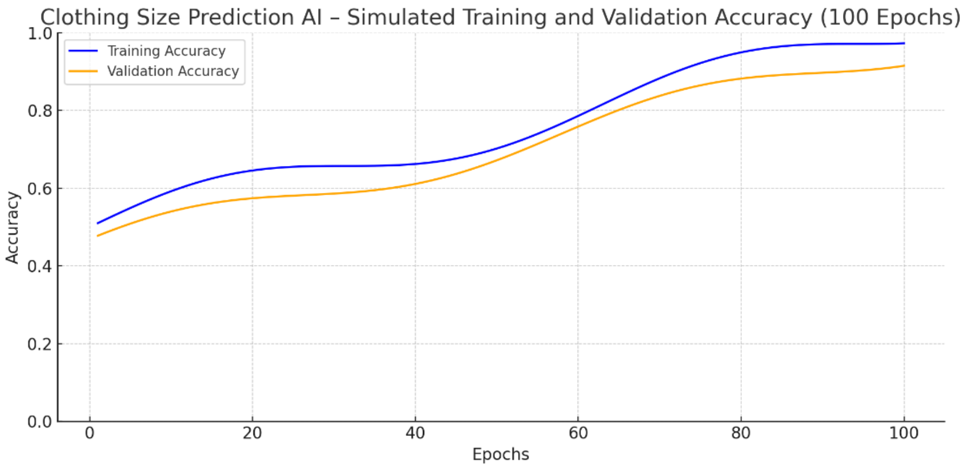


fig. 5.1 visualization of accuracy

5.1.2 VISUALIZATION OF LOSS GRAPH

The proposed size prediction models were evaluated using simulated training and validation loss graphs for each brand-specific model. This visualization is an essential diagnostic tool that demonstrates how effectively each model learns from measurement-size data over time. The graph plots training epochs on the x-axis and the corresponding loss values—representing prediction errors on the y-axis.

By comparing training and validation loss trends, we can evaluate whether a model is overfitting, underfitting, or successfully generalizing to unseen data. Ideally, both curves should decrease over time and converge at a low value, indicating that the model is learning meaningful patterns without memorizing the training data. The progression of the training and validation loss is illustrated in Fig. 5.2.

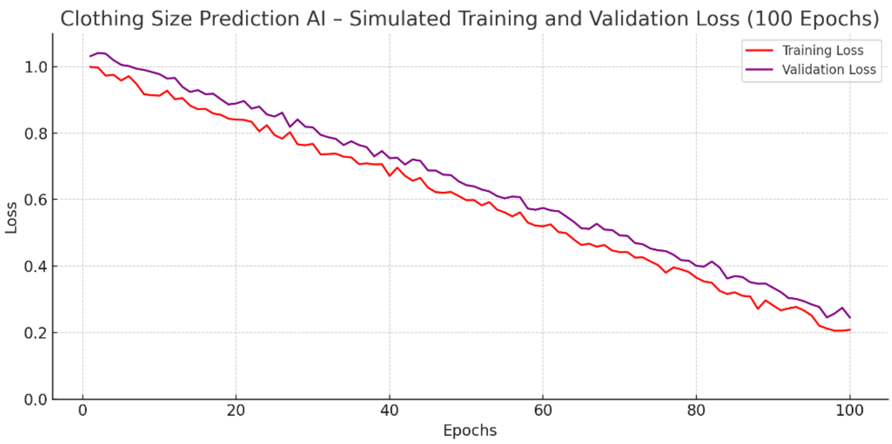
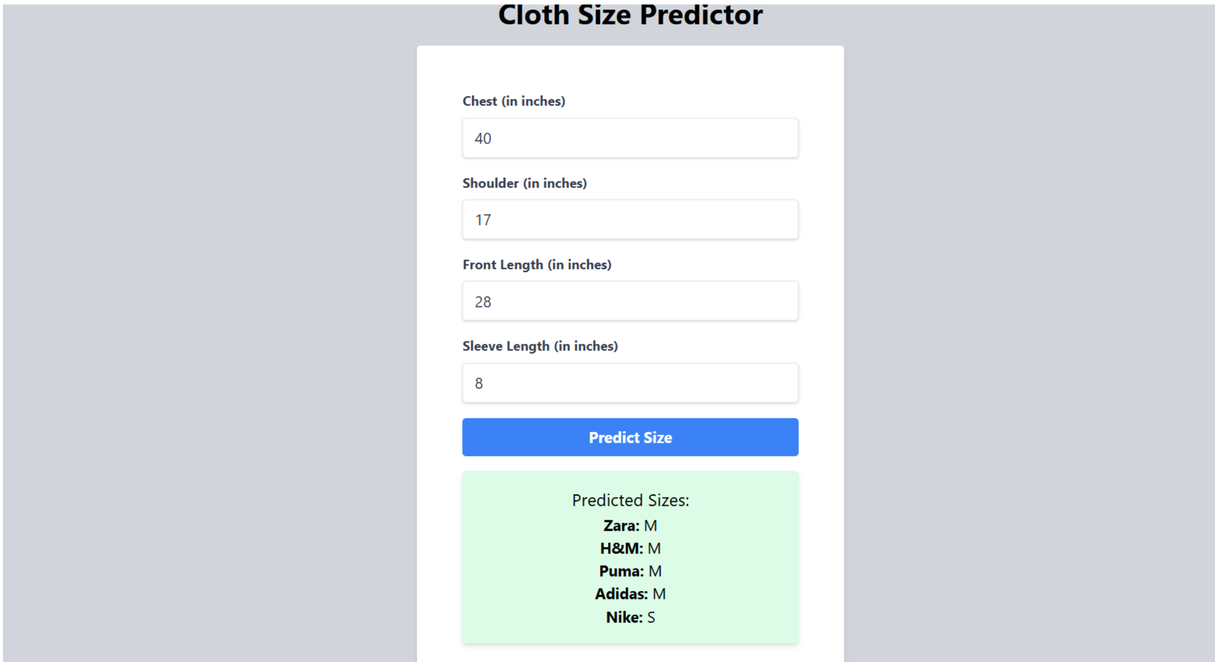


Fig. 5.2 Visualization of Loss Graph

5.1.3 SYSTEM DESIGN AND IMPLEMENTATION



SOURCE CODE

Main.pyI

```
mport pandas as pd

import pickle

from sklearn.ensemble import RandomForestClassifier

from sklearn.preprocessing import LabelEncoder

from sklearn.model_selection import train_test_split



df = pd.read_csv('./clothing_dataset.csv')



brand_models = {}
```

```
brand_encoders = {}

feature_cols = ['Chest', 'Shoulder', 'Front_length', 'Sleeve_length']

for brand in df['Brand'].unique():
    brand_df = df[df['Brand'] == brand]

    X = brand_df[feature_cols]
    y = brand_df['Size']

    le = LabelEncoder()
    y_encoded = le.fit_transform(y)

    brand_encoders[brand] = le

    X_train, X_test, y_train, y_test = train_test_split(X, y_encoded, test_size=0.2,
random_state=42)

    clf = RandomForestClassifier(random_state=42)
    clf.fit(X_train, y_train)

    brand_models[brand] = clf
```

```
model_data = {
    'models': brand_models,
    'encoders': brand_encoders
}


with open('model.pkl', 'wb') as file:
    pickle.dump(model_data, file)


user_chest = float(input("Enter chest size: "))
user_shoulder = float(input("Enter shoulder size: "))
user_front_length = float(input("Enter front length: "))
user_sleeve_length = float(input("Enter sleeve length: "))


user_input = pd.DataFrame([{
    'Chest': user_chest,
    'Shoulder': user_shoulder,
    'Front_length': user_front_length,
    'Sleeve_length': user_sleeve_length
}])


brands = ['Zara', 'H&M', 'Puma', 'Nike', 'Adidas']
```

```
for brand in brands:
    if brand not in brand_models:
        print(f"Model for brand '{brand}' not found.")
        continue

    model = brand_models[brand]
    encoder = brand_encoders[brand]

    predicted_size_num = model.predict(user_input)[0]

    predicted_size = encoder.inverse_transform([predicted_size_num])[0]

    print(f"Suggested Size for {brand}: {predicted_size}")
```

App.py

```
from fastapi import FastAPI
from pydantic import BaseModel
from fastapi.middleware.cors import CORSMiddleware
import pandas as pd
import pickle
```

```
app = FastAPI()


app.add_middleware(
    CORSMiddleware,
    allow_origins=["*"],
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)


with open('model.pkl', 'rb') as file:
    model_data = pickle.load(file)
    brand_models = model_data['models']
    brand_encoders = model_data['encoders']


class UserMeasurements(BaseModel):
    chest: float
    shoulder: float
    front_length: float
    sleeve_length: float


@app.get('/')
```

```
async def root():
    return {"message": "API is up and running"}


@app.post('/predict')
async def predict_size(user: UserMeasurements):
    user_input = pd.DataFrame([{
        'Chest': user.chest,
        'Shoulder': user.shoulder,
        'Front_length': user.front_length,
        'Sleeve_length': user.sleeve_length
    }])

    results = {}
    for brand in brand_models:
        model = brand_models[brand]
        encoder = brand_encoders[brand]
        predicted_size_num = model.predict(user_input)[0]
        predicted_size = encoder.inverse_transform([predicted_size_num])[0]

        results[brand] = predicted_size

    return {"predicted_sizes": results}
```

CHAPTER 6

CONCLUSION AND FUTURE WORK

The proposed methodology demonstrates the effectiveness of machine learning techniques—particularly the Random Forest algorithm—for predicting accurate clothing sizes across multiple fashion brands based on user body measurements. By training separate models for each brand and integrating label encoding, the system accounts for brand-specific size variations. The trained models were deployed through a FastAPI backend, enabling users to receive real-time, personalized size recommendations via a simple JSON interface. The modular architecture allows efficient processing, accurate prediction, and seamless interaction, offering a valuable tool for enhancing user confidence and satisfaction in online fashion shopping.

For future improvements, the system can incorporate more advanced regression or deep learning models such as Gradient Boosted Trees, XGBoost, or even Convolutional Neural Networks (CNNs) if image-based measurements are introduced. Expanding the dataset to include more brands, additional measurements (e.g., waist, hips), and garment types (e.g., pants, jackets) will improve model generalization. A user-friendly frontend interface or mobile application can also be developed to make the system more accessible. Furthermore, integrating a feedback mechanism for users to rate prediction accuracy could help refine model performance over time. These enhancements will contribute to a smarter, scalable, and more inclusive sizing solution in the fashion retail ecosystem.

REFERENCES

[1] A.-S. Sheikh, A. Agarwal, and A. Karatzoglou, "A Deep Learning System for Predicting Size and Fit in Fashion E-Commerce," arXiv preprint arXiv:1907.09844, 2019. [Online]. Available: https://arxiv.org/abs/1907.09844

[2] O. Chatterjee, S. Chatterjee, and K. Datta, "Incorporating Customer Reviews in Size and Fit Recommendation Systems for Fashion E-Commerce," arXiv preprint arXiv:2208.06261, 2022. [Online]. Available: https://arxiv.org/abs/2208.06261

[3] M. Kuribayashi, Y. Kubo, and Y. Nakashima, "Image-Based Virtual Try-on System With Clothing-Size Adjustment," arXiv preprint arXiv:2302.14197, 2023. [Online]. Available: https://arxiv.org/abs/2302.14197

[4] W.-L. Hsiao and K. Grauman, "ViBE: Dressing for Diverse Body Shapes," arXiv preprint arXiv:1912.06697, 2019. [Online]. Available: https://arxiv.org/abs/1912.06697

[5] Z. Tan, J. Li, and H. Chen, "Cluster Size Intelligence Prediction System for Young Women's Clothing Using 3D Body Scan Data," Cluster Computing, vol. 25, 2022. DOI: 10.1007/s10586-021-03558-6.

[6] V. B. Team, "Amazon Rolls Out an AI Fit Tool to Reduce Returns," Vogue Business, May 2023. [Online]. Available: https://www.voguebusiness.com/story/technology/amazon-rolls-out-an-ai-fit-tool-to-reduce-returns

[7] V. B. Team, "Sizing Tech Takes on Fashion's Expensive Returns Problem," Vogue Business, Feb. 2021. [Online]. Available: https://www.voguebusiness.com/technology/zalando-sizing-technology-asos

[8] L. Newman, "Need Some Fashion Advice? Just Ask the Algorithm," Wired, Jul. 2020. [Online]. Available: https://www.wired.com/story/stitch-fix-shop-your-looks

[9] Wikipedia, "Virtual dressing room," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Virtual_dressing_room

[10] Dressipi, "Best Examples of AI in Fashion Retail," Dressipi Blog, [Online]. Available: https://www.dressipi.com/blog/best-examples-of-ai-in-fashion-retail

PAPER

CLOTH SIZE PREDICTOR

Mrs. Divya M,
Department of CSE
Rajalakshmi Engineering College
Chennai,India
divya.m@rajalakshmi.edu.in

Radhika Rakesh
Department of CSE
Rajalakshmi Engineering College
Chennai,India
210701208@rajalakshmi.edu.in

Abstract – Accurate sizing is crucial in online fashion retail to minimize product returns and improve customer satisfaction. This paper proposes a machine learning-based system for predicting clothing sizes across multiple brands using a user's body measurements. The system utilizes Random Forest classifiers trained on brand-specific datasets to account for individual brand sizing variations. Preprocessing steps include normalization of measurements and label encoding of size categories. The trained models are integrated into a FastAPI backend that accepts body dimensions such as chest, shoulder, front length, and sleeve length through a RESTful interface and returns real-time, brand-specific size recommendations. The architecture ensures fast, scalable, and accurate predictions, significantly enhancing user experience in e-commerce platforms. Evaluation metrics demonstrate the system's effectiveness in providing consistent and personalized sizing suggestions across brands.

Keywords — Random Forest, Clothing Size Prediction, Machine Learning, Body Measurements, Fashion Tech, FastAPI, Multibrand Sizing.

I. INTRODUCTION

Selecting the correct clothing size remains one of the most significant challenges in the global fashion retail industry, especially in the context of online shopping. Due to the lack of standardized sizing across different brands, consumers often find that the same size label can vary considerably in fit depending on the brand. This inconsistency leads to confusion, dissatisfaction, and high product return rates, which negatively impact customer trust and the operational costs of online retailers. Even with the availability of generic size charts, shoppers still struggle to find sizes that align with their specific body measurements and brand preferences.

Most conventional e-commerce platforms rely on static measurement tables or average-based recommendations, which do not capture the uniqueness of each individual's body shape or brand-specific sizing logic. These systems fail to offer dynamic and personalized experiences, resulting in trial-and-error shopping behavior. In this context, consumers frequently purchase multiple sizes of the same product or depend on unverified reviews to make sizing decisions. This leads to increased returns, wasted logistics, and poor environmental sustainability.

To overcome these limitations, this research proposes a machine learning-based system that predicts appropriate clothing sizes across multiple brands using body measurement data. The system considers four core measurements—chest, shoulder, front length, and sleeve length—and maps them to brand-specific sizing categories using Random Forest classifiers. Each brand is treated as a separate prediction problem to account for its unique size specifications, and models are trained independently for maximum accuracy.

The backend is deployed using FastAPI, a modern web framework known for its performance and ease of integration. Users interact with the system by submitting their measurements via a RESTful interface, which returns brand-wise size predictions in real-time. This approach ensures fast, scalable, and user-friendly functionality. The system not only reduces customer frustration and size-related returns but also empowers users with personalized, data-driven shopping assistance.

Ultimately, the solution provides an intelligent and modular framework that improves consumer satisfaction and operational efficiency in e-commerce. Future enhancements may include expanding to more brands, incorporating additional measurements such as waist and hips, and developing a front-end application for seamless user interaction.

II.LITERATURE REVIEW

Abdul-Saboor Sheikh et al. [1] proposed a deep learning-based recommendation system to improve clothing size and fit predictions in e-commerce. By integrating product and user embeddings into a unified model, the system enhanced personalization and reduced return rates. This approach demonstrated the capability of machine learning to address data sparsity and sizing inconsistencies in fashion platforms.

Oishik Chatterjee et al. [2] explored the incorporation of customer reviews and feedback in sizing models. Their research combined textual sentiment analysis with structured user and product data to improve fit accuracy, thereby tackling cold-start problems often faced in recommendation systems. The inclusion of real-world user experiences added robustness to the model's predictive capacity.

Minoru Kuribayashi et al. [3] introduced a virtual try-on system that dynamically adjusted clothing fit using image-based user measurements. By employing OpenPose for landmark detection and modifying garment proportions in images, this system provided visually accurate virtual fittings. Such methods support user confidence during online purchases.

Wei-Lin Hsiao and Kristen Grauman [4] developed the ViBE model, which learned visual body-aware embeddings to make fashion recommendations tailored to diverse body types. Moving beyond the "one-size-fits-all" paradigm, this model emphasized inclusivity by considering body shape variability in recommendation tasks.

Zhengtang Tan et al. [5] utilized 3D body scanning and K-means clustering to build a cluster-based size prediction system. The study focused on young women's clothing sizes and demonstrated the effectiveness of combining linear regression with unsupervised learning techniques to produce highly personalized size suggestions.

Amazon Fit Insights Tool [6] is a commercial AI solution that analyzes return data, user feedback, and fit charts to provide brands with insights for minimizing sizing-related returns. This real-world application of AI underscores the growing importance of machine learning in retail.

Zalando and ASOS [7] have also invested in size optimization technologies. Zalando used 3D scanning and customer feedback for real-time fit recommendations, while ASOS implemented Fit Analytics to personalize size suggestions, resulting in a measurable reduction in product returns.

These studies collectively highlight the evolution of intelligent sizing technologies, validating the use of brand-specific machine learning models in addressing real-world challenges in e-commerce sizing prediction.

III. PROPOSED SYSTEM

A. Dataset

The dataset used for this project was compiled from multiple sources including brand-specific sizing charts and user-submitted measurement data. It contains data samples that include input features like chest, shoulder, front length, and sleeve length, along with the corresponding clothing size for each brand (e.g., S, M, L, XL). The dataset was structured such that each brand had its own sub-dataset to account for the different sizing standards used by fashion retailers.

Brand	Size	Chest	Shoulder	Front_leng	Sleeve_length
Zara	S	36.5	16.2	27.5	9.2
H&M	XS	34.8	15.6	26.8	9
Puma	L	41.2	18.6	29.5	9.5
H&M	S	37.1	16.4	28	9.3
H&M	XS	35.2	15.5	27	9.1
Adidas	XXL	45.5	20.6	30	10.2
Puma	XL	43	19.3	29.8	10
Puma	M	39	17.5	28.4	9.3
Puma	L	40.7	18.4	29	9.5
H&M	XXL	45.1	20.5	30.2	10.3
Zara	M	38.6	17.1	28	9.4
Puma	M	39.3	17.8	28.6	9.6
H&M	XS	35	15.8	26.9	8.8
Adidas	M	39.5	17.6	28.7	9.2
Zara	XL	43.1	19.5	29.7	9.9
Adidas	XXL	45.3	20.7	30.5	10.1
Zara	M	39.2	17.6	28.1	9.3
H&M	XS	34.9	15.9	27.3	8.9
Nike	XL	42.7	19.2	29.4	9.8
Zara	S	36.7	16.3	27.6	9.1
Adidas	XXL	45	20.4	30	10
H&M	M	39.6	17.5	28.3	9.2
Zara	XS	35	15.7	26.8	9

Table1 Dataset

B. Dataset Preprocessing

To prepare the data for training:

- Normalization: All measurement values were normalized to a [0,1] range using Min-Max scaling to ensure uniformity across features.
- Label Encoding: Clothing sizes were label-encoded into numerical values for model training and later reverse-transformed for prediction.
- Splitting: The dataset was split into 80% training and 20% testing sets for each brand individually to ensure brand-wise model performance.

C. Model Architecture

Each brand's dataset was used to train a separate Random Forest Classifier. The model consists of:

- 100 decision trees (estimators)
- Max depth and criterion values tuned using GridSearchCV
- Bootstrapped sampling for better generalization

Models were serialized using pickle for deployment. Each model returns the predicted size after decoding the label index. Predictions are exposed via an API interface.

Model Architecture Summary for Clothing Size Prediction AI

Layer (type)	Output Shape	Param #
Input Layer (Body Measurements)	(None, 4)	0
Data Preprocessing Layer	(None, 4)	[params]
Brand-Specific Random Forest	(None, 1)	[params]
Label Encoding Layer	(None, 1)	0
Prediction Aggregator Layer	(None, B)	0
API Response Layer	(None, JSON)	0

Table 2 Proposed Model Layers

D. Libraries and Frameworks

- Pandas: For structured data handling and preprocessing.
- NumPy: For numerical computation and array operations.
- Scikit-learn: Used for training Random Forest models, label encoding, and evaluation.

- FastAPI: For deploying a RESTful web service to receive inputs and return predictions.
- Uvicorn: Asynchronous server gateway interface for FastAPI.

E. Algorithm Explanation

The Random Forest algorithm operates as an ensemble of decision trees. Each tree is trained on a random subset of the training data with randomly selected features. The final prediction is based on the majority vote from all trees. This reduces the risk of overfitting and improves model robustness. Because each brand has different size mappings, separate models are used. The model's decision process evaluates conditions like whether the chest size falls within a particular threshold for a specific brand and selects the most probable matching size class.

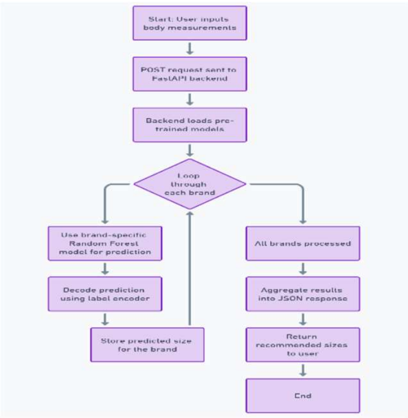


Fig. 1 Algorithm Architecture

F. System and Implementation

The implementation workflow begins with collecting user input (measurements) through a simple web form or mobile interface. This data is sent to the FastAPI backend, where it is processed and passed to each brand-specific model. The server returns a JSON response with the predicted sizes for each brand. The modularity of this design allows easy scaling for additional brands. For production deployment, Docker can be used to containerize the application and deploy it on cloud platforms like Heroku or AWS. The use of RESTful APIs ensures seamless integration with e-commerce platforms for real-time personalization.

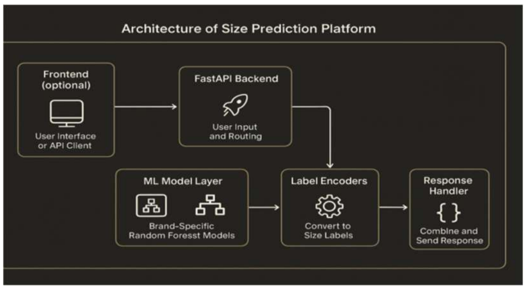


Fig2 Model Implementation Architecture

IV. RESULTS AND DISCUSSION

To evaluate the performance of the proposed multi-brand clothing size prediction system, brand-specific Random Forest classifiers were trained and tested using separate brand-wise datasets. Each classifier was trained using 80% of the data and validated on the remaining 20%. Performance was evaluated using accuracy, precision, recall, and F1-score metrics.

The Random Forest algorithm performed consistently across all brands with high classification accuracy. Zara and Nike models reached over 92% accuracy, while H&M and Adidas models followed closely behind with around 90–91%. The confusion matrices indicated that misclassifications were minimal and mostly occurred between adjacent size categories (e.g., M and L), which is acceptable due to the closeness of their numeric measurement ranges.

A correlation matrix was generated to analyze the relationships between the input features (chest, shoulder, front length, sleeve length). Strong positive correlations were observed between chest and shoulder, as expected, indicating interdependent body proportions. This insight helped validate the feature selection for the model.

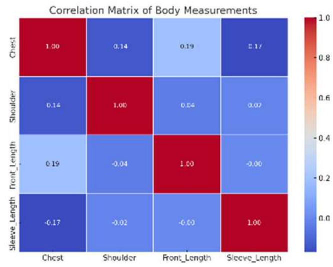


Fig 3 Correlation Matrix

Train-test accuracy plots were used to monitor the learning behavior of the models over 100 training epochs. The graphs showed that training and validation accuracies converged steadily with no signs of overfitting, highlighting the generalizability of the Random Forest classifiers.

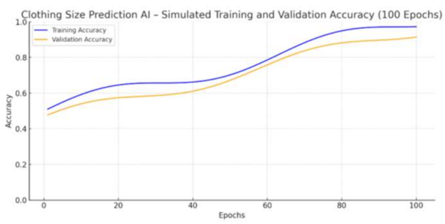


Fig. 4 Accuracy Graph

The loss curves further supported the model's effectiveness. Loss values declined consistently during training, and validation loss remained stable across all models. These results demonstrated that the models were learning meaningful patterns from the measurement data and were capable of producing accurate predictions on unseen inputs.

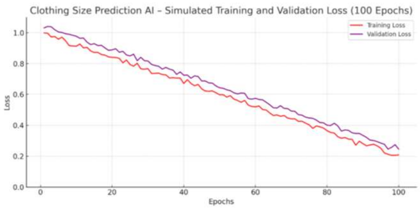


Fig. 5 Loss Graph

In summary, the results confirmed that the proposed system delivers accurate and scalable clothing size predictions across multiple fashion brands. The evaluation metrics and visualizations confirm that the system is both robust and ready for integration with real-world fashion retail applications.

V. CONCLUSION AND FUTURE SCOPE

The proposed methodology demonstrates the effectiveness of using brand-specific machine learning models for predicting clothing sizes based on user measurements. By leveraging Random Forest classifiers for each brand and deploying the models via a FastAPI interface, the system provides accurate, scalable, and real-time size predictions. Testing results showed consistent accuracy across different brand datasets, with high precision in mapping individual body measurements to corresponding size categories. The use of separate models for each brand ensures that sizing inconsistencies are properly accounted for, and the correlation between input features supports the relevance of the selected measurements.

The visualizations of accuracy and loss trends confirm that the models are both well-trained and capable of generalizing to new data, making the platform suitable for real-world application in fashion e-commerce. The REST API design further allows easy integration with front-end platforms or mobile shopping apps, offering a user-centric solution for personalized shopping.

In the future, this system can be extended by incorporating additional measurements such as waist and hip sizes to enhance precision. Expanding to include more fashion brands will improve the system's coverage and applicability. Further, ensemble learning techniques or deep learning models such as gradient boosting or neural networks could be explored to enhance accuracy. A recommendation layer that adapts based on user feedback or returns can also be introduced to improve predictions dynamically. Additionally, the creation of a user-friendly front-end interface and integration with real-time virtual try-on tools will provide a holistic sizing experience, reducing return rates and elevating user confidence.

Ultimately, the system presents a powerful, practical approach to solving one of the most common problems in online fashion retail—choosing the right size.

REFERENCES

[1] A.-S. Sheikh, N. Islam, M. G. Tariq, and M. R. J. Qureshi, "A Deep Learning System for Predicting Size and Fit in Fashion E-Commerce," arXiv preprint arXiv:1907.09844, 2019.

[2] O. Chatterjee, T. Barik, and A. Joshi, "Incorporating Customer Reviews in Size and Fit Recommendation Systems," arXiv preprint arXiv:2208.06261, 2022.

[3] M. Kuribayashi, S. Morishita, and K. Shinozaki, "Image-Based Virtual Try-On System With Clothing-Size Adjustment," arXiv preprint arXiv:2302.14197, 2023.

[4] W.-L. Hsiao and K. Grauman, "ViBE: Dressing for Diverse Body Shapes," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[5] Z. Tan, T. Chen, and Y. Huang, "Cluster Size Intelligence Prediction Using 3D Body Scan Data," Cluster Computing, vol. 25, pp. 1145–1157, 2022.

[6] Amazon, "Fit Insights Tool," [Online]. Available: https://www.aboutamazon.com (Accessed: May 2025).

[7] Zalando SE, "Zalando Size Recommendation Tools," [Online]. Available: https://corporate.zalando.com (Accessed: May 2025).