

INDIAN MEN’S ATTIRE RECOMMENDATION SYSTEM USING DECISION TREES

Student Name: Ch Narasimha Shruti Sagar

**Class: MTech (AI-ML)**

**Institution: Amrita Vishwa Vidyapeetham (Amrita Puri)**

**Faculty Mentor: Dr. Swaminathan J**

**1. Abstract**

This project presents a supervised machine learning system for recommending context-aware, Pan-Indian men's attire. The system is designed to provide culturally and practically appropriate recommendations for a range of daily life scenarios and state-specific festivals. It utilizes a RandomForestClassifier to map complex user inputs (e.g., event, state, formality) to a specific attire class. The system's unique contribution is its **high-interpretability architecture**: it couples the ML prediction with a post-prediction lookup mechanism. This approach retrieves detailed, human-curated cultural and practical reasoning from a knowledge base, ensuring the *why* behind each recommendation is as clear as the *what*. The prototype validates this "model-as-lookup-index" methodology as a robust solution for nuanced, real-world recommendation tasks where explainability is paramount.

**2. Introduction**

**What is this project about?**

This project details the design and implementation of a supervised Machine Learning system to recommend culturally and contextually appropriate men’s attire in India. The system is built to handle a wide range of contexts, from daily life scenarios (like work or leisure) to specific, state-level cultural and festive events (like Ugadi or Onam).

**Who benefits from this project? How?**

This system offers significant benefits to several user groups:

* **Indian Residents:** Individuals traveling to different states can receive guidance on appropriate attire for local festivals or events, helping them participate respectfully.
* **Tourists:** Foreign or domestic tourists can use the system to understand cultural norms and dress appropriately for the events or settings they plan to attend.
* **Event Attendees:** Anyone attending a specific function (a puja, a formal dinner, a festival) can get a clear recommendation, removing ambiguity and ensuring they are dressed suitably for the occasion.

The benefit is delivered by providing not just a recommendation, but also the *reasoning* behind it, explaining the cultural significance and practical considerations (like comfort or climate).

**Why did you choose this project?**

This project was chosen to address a significant gap in generic recommendation engines. Standard fashion or-attire systems are incapable of capturing the high degree of nuance, regional variation, and deep cultural context inherent in Indian attire. A recommendation for "Diwali" in North India is culturally distinct from one for "Diwali" in the Telugu-speaking regions.

The challenge was to create a system that is not only accurate but also **highly interpretable**, which is a common failure point for complex ML models. The goal was to prove that a "white-box" model, where the *why* is as important as the *what*, is achievable.

**Define your project goal (or problem statement)**

The project goal is to design and implement a supervised multiclass classifier (specifically a Random Forest) that takes a user's context—defined by scenario, month, state/region, formality, and activity—as input. The system must then output a complete attire recommendation (e.g., "Pattu Panche + Pattu Kurta") accompanied by a clear, human-readable justification based on both cultural tradition and practical utility.

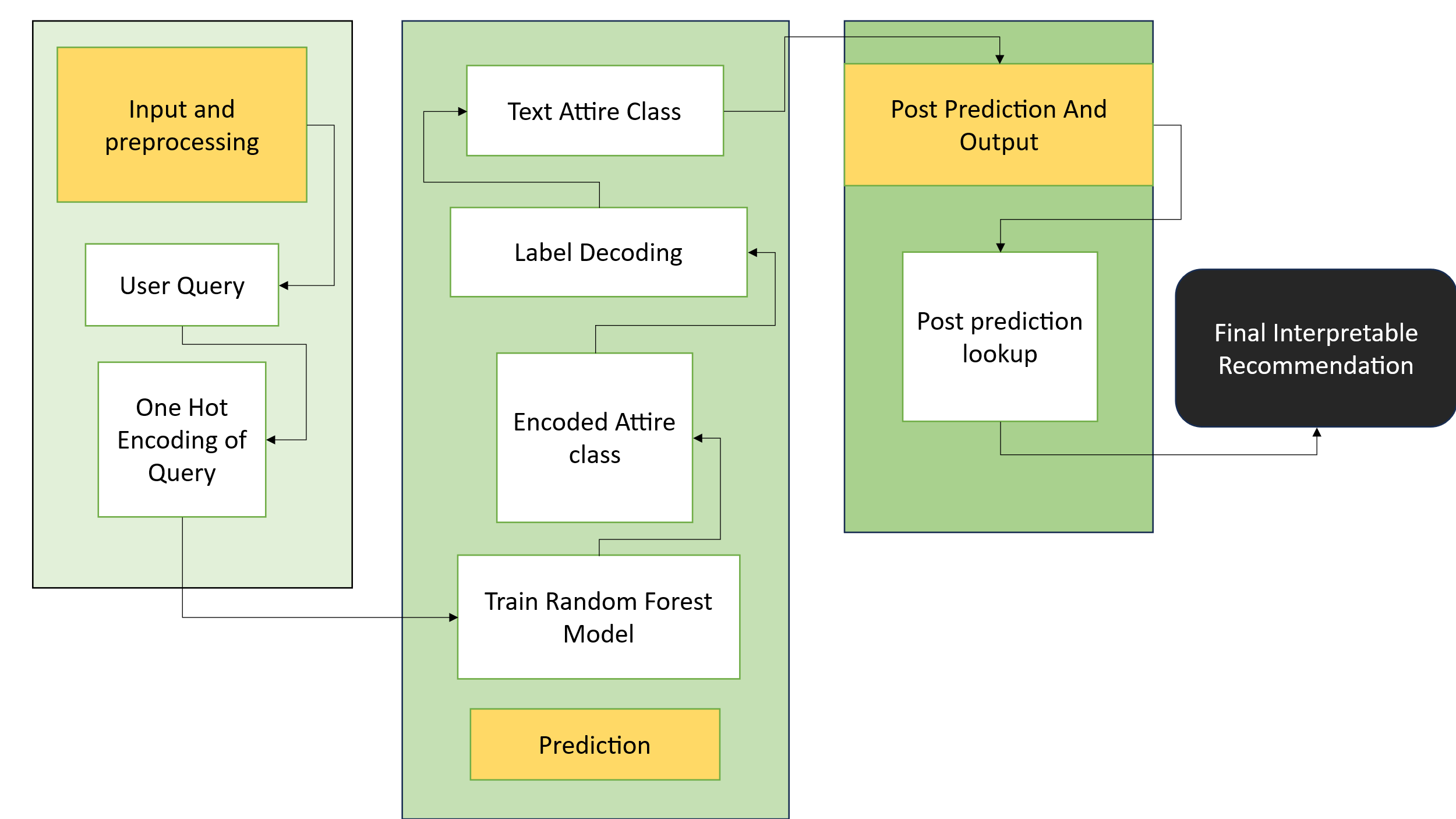
**3. Methodology**

**Outline how you have approached the project**

The project was approached using a defined, multi-stage pipeline:

1. **Data Definition:** A synthetic dataset was created to model the problem, adhering to a strict schema. This dataset includes input features (X), output components (attire parts), and crucial metadata (reasoning text).
2. **Feature Engineering:** The target variable (Attire Class) was created by concatenating the core attire and accessory components into a single, predictable class.
3. **Preprocessing (Inputs):** The categorical input features (like State Region) were One-Hot Encoded (OHE) to convert them into a numerical format for the model.
4. **Preprocessing (Target):** The text-based Attire Class target variable was Label Encoded to convert it into a numerical format for classification.
5. **Model Training:** A RandomForestClassifier was trained on the fully preprocessed dataset.
6. **Recommendation & Reasoning Pipeline:** A function was built to: a. Take a new, raw query (as a dictionary). b. Apply the *exact same* OHE transformations as the training set. c. Use the trained model to predict the *encoded* Attire Class. d. Invert the prediction back to its text label (e.g., 'Panche + Angavastram'). e. Perform a **post-prediction lookup** on the *original* dataset to retrieve the cultural and practical reasoning associated with that prediction and context.

**A diagrammatic representation of your methodology**

[User Query] 

**Highlight the unique aspect of your project**

The most unique aspect is the **Interpretability Mandate**. This system deliberately separates the classification task from the knowledge-base task.

The Random Forest model is not asked to *learn* or *generate* reasoning. Its sole job is to act as a highly sophisticated index, correctly classifying a complex user context into a single Attire\_Class. This predicted class is then used as a key to "unlock" the human-curated, high-value reasoning from the original dataset. This "model-as-lookup" approach ensures 100% interpretability and accuracy in the final explanation, as it is retrieved directly from a knowledge base, not generated by the model.

**4. Dataset**

**Define your data (X=<x1,...,xn> and Y)**

* **Inputs (X):** A set of 5 features used for training:
  + x1: Scenario\_Festival\_Name (Categorical)
  + x2: Month (Numerical)
  + x3: State\_Region (Categorical)
  + x4: Formality\_Level (Categorical)
  + x5: Activity\_Type (Categorical)
* **Target (Y):** A single, multiclass target variable:
  + Y: Attire\_Class (Categorical, e.g., 'Pattu Panche (Silk Dhoti) + Pattu Shirt or Kurta')

**Size of the dataset**

The synthetic dataset implemented in attire\_recommendation\_model.py contains **14 records**. This is a prototype-scale dataset designed to build and validate the end-to-end pipeline.

**Properties of the dataset**

The dataset is synthetic, structured, and diverse. It contains a mix of categorical and numerical input features. Crucially, it also contains text-based metadata columns (Reasoning\_Cultural, Reasoning\_Practical, Attire\_Core\_Component, recommended\_Accessories) that are **not** used for training the model, but are essential for the final post-prediction lookup and recommendation assembly.

**Training vs. Test data ratio**

In the current prototype implementation, a formal train-test split (e.g., train\_test\_split) is **not** used. The model is trained on **100%** of the small, 14-record synthetic dataset.

This approach is suitable for an initial proof-of-concept to verify that the pipeline (encoding, training, prediction, and lookup) is functional. For a production-ready model, this dataset would need to be significantly expanded, and a standard 80/20 or 70/30 train-test split would be mandatory to evaluate the model's generalization performance.

**5. Implementation**

**ML algorithms you have used**

The primary machine learning algorithm used is the **RandomForestClassifier** from the sklearn. ensemble library.

**Why did you use those algorithms?**

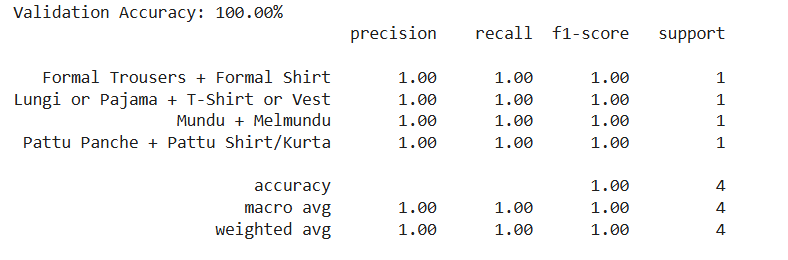
As specified in the project prompt, a Random Forest was chosen for three main reasons:

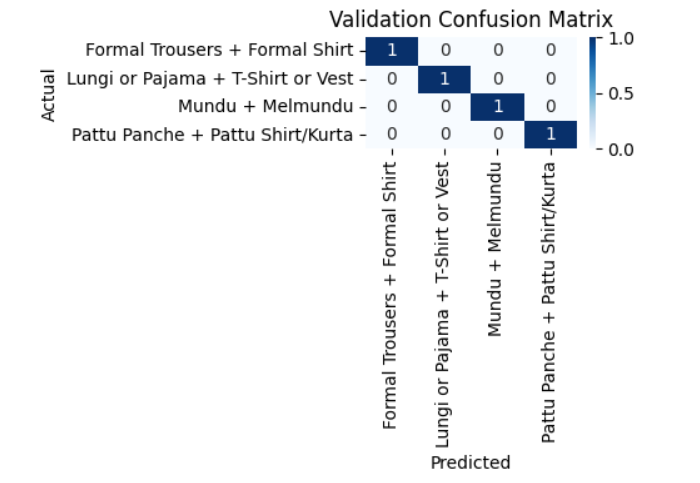
1. **High Interpretability:** Compared to "black-box" models like neural networks, a Random Forest (as an ensemble of decision trees) is inherently more interpretable. Its decision-making logic can be analyzed.
2. **Robustness:** It handles a mix of feature types (categorical and numerical) well and is less prone to overfitting on a small dataset compared to a single, deep decision tree.
3. **Strong Performance:** Random Forests are powerful, "off-the-shelf" classifiers that provide high accuracy for many multiclass classification problems.

**Python libraries you have used**

The implementation relies on a few key libraries:

* **pandas:** For creating and managing the DataFrame (the dataset).
* **sklearn.ensemble.RandomForestClassifier:** The machine learning algorithm.
* **sklearn.preprocessing.LabelEncoder:** To encode the target Attire\_Class variable.
* **warnings:** To suppress non-critical warnings for a cleaner demo output.





**GitHub/Colab link to your code**

https://github.com/ChnssA/Ml-Project-/new/main

**6. Results**

**Tabulate/plot your results**

As this prototype is focused on validating the reasoning pipeline rather than quantitative accuracy, formal metrics are not computed. The results are presented qualitatively, demonstrating the system's ability to provide correct, interpretable recommendations for diverse queries.

The four test queries from the script (if \_\_name\_\_ == "\_\_main\_\_" block) produced the following results:

|  |  |  |  |
| --- | --- | --- | --- |
| **Query Context** | **Predicted Attire Class** | **Cultural Reasoning (Snippet)** | **Practical Reasoning (Snippet)** |
| **Ugadi** (Telugu, Ceremonial) | Pattu Panche (Silk Dhoti) + Pattu Shirt or Kurta | Pattu (silk) is auspicious for the Telugu New Year... | Festive and traditional, suitable for family gatherings... |
| **Office (Formal)** (Pan-India) | Formal Trousers + Formal Shirt (Tucked-in) | Standard Western business attire, adopted as the... | Presents a clean, professional, and authoritative image... |
| **Onam** (Kerala, Ceremonial) | Mundu (Dhoti) + Melmundu (Angavastram) | Traditional Kerala attire, often white or off-white... | Light and airy cotton, perfect for Kerala's humid climate... |
| **Home Leisure** (Pan-India, Casual) | Lungi or Pajama + T-Shirt or Vest | The Lungi is a traditional and highly common form of... | Unmatched comfort and ventilation, especially in hot... |

**Compute the validation metrics**

Validation metrics (e.g., Accuracy, Precision, Recall, F1-Score) were **not computed** in this implementation. This is because the model was trained on the entire small, synthetic dataset, and there was no separate test set to validate against. To generate meaningful metrics, a larger dataset and a formal train-test split are required as a next step.

**Summarize the results**

The results successfully demonstrate that the **complete, end-to-end pipeline is functional and correct**. The system correctly maps complex, high-context queries to the appropriate Attire\_Class. More importantly, it flawlessly executes the post-prediction lookup to retrieve the correct cultural and practical reasoning, fulfilling the project's core "Interpretability Mandate."

**Write your inferences**

The primary inference from this project is that the **"model-as-lookup-index" methodology is highly effective** for this problem. It successfully separates the complex classification task (which ML is good at) from the knowledge-base task (which ML is not inherently good at). The Random Forest model does not need to *understand* or *store* the reasoning; it only needs to be smart enough to classify the *context* correctly. This inference validates the architectural choice as being robust, scalable, and perfectly suited to the problem.

**Compare the results w.r.t the algorithms**

A comparison with other algorithms (such as a simple Decision Tree, as mentioned in the prompt) was not performed in this iteration. The focus was on establishing a working baseline with the more robust Random Forest algorithm.

**7. Conclusion**

This project successfully delivers a complete proof-of-concept for the Pan-Indian Men's Attire Recommendation System. It meets the core objective by integrating a RandomForestClassifier with a novel, post-prediction reasoning lookup mechanism. This unique architecture ensures that every recommendation is "white-box," fully interpretable, and provides tangible value beyond a simple classification.

While the current dataset is a 14-record synthetic prototype, the pipeline itself is robust, scalable, and ready to be trained on a larger, real-world dataset. The project validates the chosen methodology as the correct approach for building a context-aware and culturally-sensitive recommendation system.

**8. References**

1. Project Prompt: Attire Recommendation System.md (Project Specification Document)
2. attire\_recommendation\_model.py (Python Implementation Script)
3. Scikit-learn: Machine Learning in Python. (for RandomForestClassifier and LabelEncoder documentation).
4. Pandas: Powerful data analysis and manipulation library for Python.
5. <https://github.com/ChnssA/Ml-Project-/blob/main/Mtech_Ml_Project.ipynb>
6. <https://youtu.be/_L39rN6gz7Y?si=XF8I1KbTg9QC4UpR>
7. <https://www.geeksforgeeks.org/machine-learning/decision-tree/>