# **A Scalable Personalization Framework Through Tree Based Clustering:**

Personalization consists of tailoring a service or a product to accommodate specific individuals, groups or segments of individuals. A personalized product recommendation isn't based on a random assumption or blind guess, it is based on some specific calculations on individuals background and/or current behavior. Clustering on the other hand is a task of finding the intrinsic structure of data by organizing data objects into similarity groups or clusters. It is a type of unsupervised machine learning because no class labels denoting a priori partition. The way these two go together in a recommendation engine setting is, we can map our current user to a precalculated group by some calculations and recommend products which is/are popular/favorable in that group.

Through this framework I have addressed 3 different types of data availability scenarios. 1. No historical data available, 2. Only historical demographic/session-based information available for a set of end-users and 3. Historical demographic/session-based information with mappable reaction/action/activity/shopping/buying behavior data available for a set of end-users.

At the first case, when no historical information available we have to fall back to Rule Based Clustering for personalization, but even so, we are still in need of a scalable way to build models for personalization which is capable of building models by implementing variety set of rules easily instead of focusing on one set of rules at a time. Keeping this in mind, I have designed the framework which takes input a collection of mutually exclusive and collectively exhaustive set of rules and build a resultant model from it with exactly one line of code. Without going into much technical details, the way it works is, given a mutually exclusive and collectively exhaustive set of rules the framework exploits the internal functionality of classic “Decision Tree” for classification algorithm to create a standardize way to create rule-based clustering model.

For the second and third data availability scenario, the framework follows the similar set of mechanisms to create the final personalization model which can be broadly categorized into a 4-step process as follows,

**Divide**: Divide the historical end-user data into numerous micro groups using the CLTree Algorithm. It uses a mechanism which is more popular in supervised learning problems named decision tree construction with some modifications.

**Assign**: Assign a representation vector to each of the micro groups created from previous step. Representation vector is a vector that represents the data in the group, it can be anything as long as it represents the group of end-users it encompasses. 3 potential choices are,

1. Average Shopping/Buying Behavior of all the members of the underlying group.
2. Average of demographic information of all the members of the underlying group.
3. Centroid of the hyper rectangle of the group.

**Conquer**: Use agglomerative clustering with a qualification criterion on representation vectors of all micro groups to bring them together (merge) to form a manageable set of qualified groups/clusters.

**Simplify**: Simplify the trained model into a Binary Search Tree which will be used at the time of generating label prediction. The goal is to make the final model as optimized as possible in terms of space and time complexity.

Glimpse of the how a final model looks like,

A close up of a map

Description automatically generated

Some of the glaring advantage of this framework other than its versatility, scalability and efficiency over traditional methods are following,

1. This framework can find clusters in the data without making prior assumptions (Ex. Number of clusters etc.) about the data.
2. It provides descriptions of the resulting groups/subgroups in terms of hyper rectangle regions.
3. It deals with the outliers effectively.

In addition to the above, this framework addresses a fundamental problem of personalization through clustering for recommendation engine. In case of third data availability scenario, it makes more sense if we find a way to put end-users in the same clusters who tend to have similar reactions to the similar recommendations rather than relying on just demographic information and also have ability to map the current user to the group where the other users tends to have similar reactions to similar recommendations based on only the available demographic information about the user. This framework provides the ability to do just that, in a scalable, sustainable and adaptable way by providing the ability of doing clustering by using two different set of data for a set of end-users. Only difference between data availability scenario second and third is how you choose to assign representation vectors to the micro groups after the divide step. If you are at third data availability scenario, you probably should assign the representation vectors to each micro group by using average Shopping/Buying Behavior of all the members of the respective underlying group to take advantage of the above-mentioned property.