

Fashion Retail Recommendation System by Multiple Clustering

Pierfrancesco Bellini, Luciano Alessandro Ipsaro Palesi, Paolo Nesi, Gianni Pantaleo
University of Florence, DINFO dept, DISIT lab (<https://www.disit.org>)
emails: <name>.<surname>@unifi.it

Abstract— Fashion retail has a large popularity and relevance, allowing customers to buy anytime finding the best offers and providing nice experiences in the shops. Consequently, Customer Relationship Management solutions have been enhanced by Information and Communication Technologies to better understand the behaviour and requirements of customers, engaging and influencing them to improve their buying experience, as well as increasing the retailers' profitability. Current solutions on marketing provide a too general approach, based on most popular or most purchased items, losing the focus on the customer centricity. In this paper, a recommendation system for fashion retail shops is proposed, based on a multi clustering approach of items and users' profiles in online and on physical stores. The proposed solution relies on association rules mining techniques, allowing to predict the purchase behavior of newly acquired customers, thus solving the cold start problems which is typical of current state of the art systems. The presented work has been developed in the context of the Feedback project founded by Regione Toscana, and it has been conducted on real retail company Tessilform, Patrizia Pepe mark. The recommendation system has been validated in store, as well as online.

Keywords—recommendation systems, clustering, customer and items clustering composed.

I. INTRODUCTION

The competitiveness of retailers strongly depends on the conquered reputation, brand relevance and on the marketing activities they carry out. The latter aspect is exploited to increase the sales and thus a retailer, through marketing, should be able to encourage customers to buy more items or more valuable items. Today consumers tend to buy more on ecommerce and the COVID-19 situation also stressed this condition. Online shopping offers the possibility to buy at any time of the day; customers buy where they find the best offer, online as well as offline, and they are also influenced by an increasing amount of information from blogs, communities, and social networks. To retain a customer is therefore an extremely difficult achievement, and in some measure can get easily out of control.

Currently, ICT (Information and Communication Technologies) offers customer relationship management (CRM) solutions that enable the recording of user data profiles, from customer information to product details, to sales transactions. CRM comprises a set of processes and enables systems supporting a business strategy to build long term, profitable relationships with specific customers [36]. Customer data and information technology (IT) tools form the foundation upon which any successful CRM strategy is built. Swift [37] defined CRM as an “enterprise approach to understanding and influencing customer behavior through meaningful communications in order to improve customer acquisition, customer retention, customer loyalty, and customer profitability”. However, CRM solutions on the market have a very traditional approach based on popular items or bundled offers, similar items or featured items and therefore often leave out the important customer centricity in any marketing strategy. In addition, there are IOT tools from

the big vendors, which promise an evolved engagement on various levels, interacting with less queues, promotions, more involvement, assistance, but which are hardly triggered within companies, especially retail ones, which need more flexible solutions. To date, therefore, market solutions are unable to build and leverage user, historical, social, and behavioral profiles. Through transactions, retailers can generate knowledge about their consumer's behavior. One of the techniques receiving more attention from researchers to generate consumer knowledge, is machine learning, specifically clustering techniques. Clustering techniques are used to group customers by similarity. So, retailers can tailor marketing actions more effectively than general marketing actions. Understanding the reasons why consumers choose a specific item within the store is important to the retailer. In addition, knowing the consumer's needs through the factors that influence shopper's decision-making process is important for the store's own growth. This is what recommendation systems are all about. Recommendation systems are applications that assist users in finding items (products, services and information) that best suit their preferences [29]. The generated recommendations are personalized, in the sense that they have been generated for a user or a group of users, or not personalized (e.g. best-selling items, or selection of items). Non-personalized recommendations are typically not addressed by research.

State of the art recommendation systems today do not solve typical retail problems. Most of the retail companies today have both online and physical store customers assisted in purchasing by shop assistants. With the GDPR (General Data Protection Regulation) rules [22], often the customer demographics are differently collected in different areas and shops where different rules are adopted. Deep learning methods to improve accuracy are hard to be adopted for the scarcity of data. For example, in fashion retail shops most are anonymous transactions of one single item, and recurrent acquisition are performed every 8-12 months, for costs and seasonality aspects. Regarding classification methods, the multichannel nature of retailers provides data with different features and with many incomplete records, and such aspects don't allow to apply most of these methods. As for clustering methods, they use RFM (Recency – Frequency – Monetary Value) [5] LTM (Life-Time-Value) [6] and demographic values as input and do not take advantage of the intuition typical especially of deep learning of customer behavior towards items. Another problem related to the fashion retail industry is the seasonality of items. The life of most items is 6 months to 1 year.

In this paper, a recommendation solution in the context of fashion retail is proposed. In order to solve the above-mentioned problems of **cold start, computational complexity, low number of returns in the shops of fashion retails, more mediated interaction in the shop and more direct online, seasonality of products**, we realized a multi clustering approach by taking as input the RFM value of online and physical stores separately. To solve the problem of

the seasonality of the products we have clustered the items. In addition, we enriched the input data with the customer behavior towards the items. To solve the cold starting problem of cluster-based recommendation systems, we have used the association rules mining technique that allows us to predict the purchase behavior of newly acquired customers. The work presented has been developed in the context of **Feedback** research and development project founded by Regione Toscana, Italy. Partners of the project are: VAR Group, University of Florence, TESSIFORM (*Patrizia Pepe* trademark), SICETELECOM, 3F CONSULTING and CONAD (External partner). The studies we are going to illustrate below were conducted on real retail company Tessiform: fashion retailer with online sales and different store in the world, mainly in Italy.

The paper is structured as follows. In Section II, related work on recommendation systems is presented. Section III described the system architecture adopted in Feedback solution. In Section IV, the proposed recommender systems based on multi clustering is presented using a number of subsections. The solution allowed to prepare the recommendations in advance and consume them in real time when the conditions occur, or for stimulating the customer to return in the shop. In Section V, the assessment and validation are reported. Conclusions are drawn in Section VI.

II. RELATED WORK

Recommendation techniques can be classified, according to the sources of knowledge they use, into six categories: [31], [23]. The **content-based approaches** recommend items by computing similarities among items and users through a set of features associated to them [39], [40]. For example, for a clothing item the considered features can be: the group (shirt, sweater, T-shirt, etc.), color, popularity, etc.; while for the users: demographic aspects, surveys answers, etc.

The **collaborative filtering-based** approach is based on the historical data of the user's interactions with the items, either explicit (e.g., user's ratings) or implicit feedback (e.g., purchase, visit). The mathematical techniques used are the neighborhood method and the latent factor model [35]. The neighborhood method identifies relationships among elements or, alternatively, among users. The latent factor model sets a number of evaluation methods to characterize both items and users and it's mainly based on the matrix factorization (for example the ratings-matrix). These approaches do not need a representation of the items, as they are based only on ratings, so they are the best recommendation systems in regard to scalability. The accuracy of recommendations increases as user interactions increase. They have cold start problems for both new items and new users. The **Demographic-based** approaches generate recommendations on the basis of the user's demographic profile (age, gender, education, etc.). They do not require a user ratings history, and they have cold start problems for new items. The **knowledge-based** approaches are based on the knowledge of item features which meet the users' needs. They do not have cold start problems, require a broad knowledge of the domain and, in case of many items, they are very difficult to implement. The **community-based** systems make recommendation through the preferences of users' friends in contexts of social networks or communities. The basic concept is that a user tends to rely on recommendations from their friends instead of those of similar but anonymous users. This approach is very useful for cold-start recommendations. The **hybrid-based** recommender

systems combine two or more recommendation approaches in different ways. Usually, considering two different approaches, the advantages of the former are used to mitigate the weakness of the latter.

The sources of knowledge are usually represented by three types of descriptors for: items, users and transactions (relations between user and item). Modern recommendation systems also use textual reviews, images and web page sequences, processed through data mining or deep learning methods, to generate recommendations.

The data mining methods for recommender systems can be summarized in three types of algorithms:

Classification. For example, the kNN classifier finds the closest k points (closest neighbors) from the training records. In [28] kNN has been implemented, with very good results in precise recommendations, to suggest short-term news to users. Decision Trees classifier works well when objects have few features, but in [27], [41] it has been shown that this technique can have low performance. In [26] it is used in the identification of target customers minimizing the recommendation errors, by selecting users to whom the recommendations should be addressed, according to which categories of purchases they have made in a selected period of time. In [28] a Naive Bayes classifier has been used to predict the user's long-term preferences in the news domain, with excellent results in accuracy. Support Vector Machines (SVM) classifier is used to find a linear hyperplane (decision boundary) that separates input data in such a way that the distance among data groups is maximized [30].

Cluster Analysis. It is the task of segmenting a heterogeneous population into a number of subgroups [1] [38]. Through Clustering it is possible to explore the data set and to organize the data for creating recommendations. Variables used in the clusters may be: Demographic [4], RFM [5], LTV [6], Demographic + RFM [7], Demographic + LTV [8], LTV+RFM [9]. The common clustering algorithms in use are: K-means (each cluster is represented by the center of the means of the data points belonging the cluster); K-Medoids (each cluster is represented by the most representative/central data point of the cluster); Clara (it is an extension to Partitioning Around Medoids, PAM, adapted to large data sets); Self-organizing map (SOM, based on artificial neurons clustering technique) [3], [33].

Association Rules aim at finding rules in the dataset that satisfy some minimum support and minimum confidence constraints. An association rule is an expression $X \Rightarrow Y$, where X and Y are item sets (e.g., Milk, Cookies \Rightarrow Sugar). Given a set of transactions T, and denoting MinSup and MinConf the minimum support and the minimum confidence constraint values, the goal of association rule mining is to find all rules having support greater than or equal to MinSup, and confidence greater than or equal to MinConf. The most common algorithms used for implementing association rule mining are Apriori [10], FP- Growth [11], SSFIM [12] and SETM [25].

In [14], a **hybrid recommendation** system combining content-based, collaborative filtering and data mining techniques has been proposed. The recommendation algorithm makes similar groups of customers using LTV value, for this the segmentation of customer based on customer behavior through RFM attributes has been performed.

With the growing volume of data acquisition, the possibility of using **deep learning** in recommendation systems have been also considered, in order to overcome the obstacles of conventional models listed above, achieving a higher accuracy of recommendation. Through deep learning it is possible to detect non-linear and non-trivial relationships among users and items from contextual, textual and visual input [19]. The main limitations of deep learning based recommendation systems are that for content-based systems there are often privacy issues in the collection of information, while for collaborative filtering the acquisition of data from different sources often results in incomplete information that greatly affects the accuracy of recommendations. The main deep learning algorithms for recommender system are: (i) Multilayer Perceptron (MLP) which is a class of feedforward artificial neural network with multiple hidden layers between the input and the output layer. [18] uses a standard MLP approach to learn interaction among user and item latent features, providing the model with flexibility and non-linearity; (ii) Autoencoders (AE) represent an unsupervised model that generate an output by compressing the input in a space of latent variables. There are many variants of autoencoders; the most common are denoising autoencoder, marginalized denoising autoencoder, sparse autoencoder, contractive autoencoder and variational autoencoder [32]. (iii) Convolutional Neural Networks (CNN) are feedforward neural networks that use convolution in place of general matrix multiplication in at least one of their layers. They can capture the global and local features and improve the efficiency and accuracy [15]. They have been used in several implementations, such as AlexNet [16] and batch-normalized Inception [17]. (iv) Recurrent Neural Networks (RNN) are typically employed to trace dynamic temporal behavior, actually in this kind of neural network the connections among the nodes form a direct graph along a temporal sequence [21]. Other fields of research have achieved an improvement by exploiting Long-Short Term Memory networks (LSTMs) that minimize RNN problems regarding the gradient vanishing/exploding. Adversary Network (AN) is a generative neural network where two neural networks are trained simultaneously within a minimax game framework [24]. Deep reinforcement learning (DRL) combines deep learning and reinforcement learning that enables to learn the best possible actions to attain the expected goals [34].

Compared to the previously discussed data mining techniques, all deep learning algorithms have cold start problems and require a considerable amount of data to improve performance. Open problems in the literature for deeplearning based recommendation systems concern the scalability of recommendation systems and the explicability of generated recommendations. On the other hand, deep learning solutions are not applicable in this case, in which the number of acquisitions per user is low.

III. SYSTEM ARCHITECTURE

In the context of fashion retail, the shops are of small size, and the customers are directly followed step by step by the attendees which provide suggestion and are ready to support them on all aspects. A similar scenario may occur on the online shopping, in which an online assistant is ready to follow the customer. In both cases, the user profiles are improved with new data in the few occasions in which the customer interact, and thus the customer might be continuously engaged with suggestions. This is obviously not possible and not

acceptable by the customers. So that, a moderated engagement tool has to be provided that may consume the possible recommendations by spending them toward the customers or the assistant a limited number of times per experience, and in specific conditions. The architecture is reported in **Figure 1**. In compliance with GDPR rules, the TOOL ADMIN stores the details of customers, items and transaction data of both stores and ecommerce website in a centralized database. Using sensors in the fitting rooms, totems in stores and RFID technology on items, customer interactions with products are stored. The recommender reads the information from the DB, generates the recommendations and stores them in the Recommendation table. The TOOL ENGAGER is responsible for sending recommendations to the customer or shop assistant within the store. After the recommendation is sent, the TOOL ENGAGER records the customer's interaction with the recommended product. The Recommender has to create a list of suggestions taking into account users' profiles and items descriptions. The recommendations have to be carefully suggested, since suggested item should not have been purchased by the customer recently, neither already proposed by the human Assistant. All the suggestions need to be generated on the basis of purchases made by the customer in the last few experiences and months, when possible. These last rules are applied directly at the final stage into the Tool Engager.

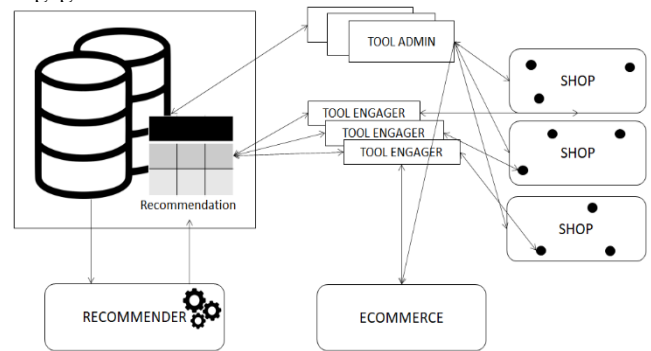


Figure 1 – General Architecture.

IV. FEEDBACK RECOMMENDER SYSTEM

The main goals consisted in increasing the customer recency, and thus to increase the number of times users contacts and sales may occur. For this purpose, the computational workflow reported in **Figure 2** has been adopted. The data are continuously collected by the administration (sales in shops: online and onsite); then a periodic clustering on items is performed. The results are taken into account in the computation of an integrated clustering driven by the user profiles and additional features to finally provide a set of suggestions of different kinds. The main steps of the workflow are described in the following subsections.

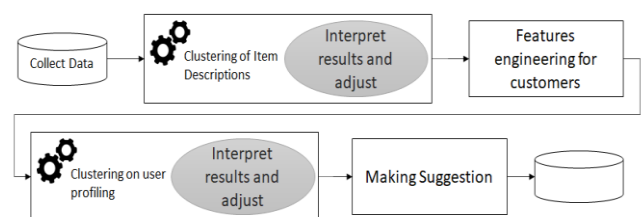


Figure 2 – General Data Computing workflow

The production of recommendations and their submission are asynchronous: (i) mediated by the assistant that may decide or not to accept and pass them to the customer, (ii) filtered by the Tool Engager according to the last actions performed by the customer, (iii) decided to be spent by sending them online via email when the time passed since the last contact with the users is greater than a reference value, etc. The produced pool of recommendations (for each potential returning user, and user kind) are generated on table, and the table can be refilled on demand or with a high rate. They include a programmed mix of *suggestions computed: by customer similarity, by items similarity, and by serendipity (randomly produced)*.

A. Clustering of Item Descriptions

As described above, the first analysis has been performed to cluster the domain of item descriptions. This allows to reduce the space and weight the relevance of item categories. In the case of fashion retail, typically the number of products is not huge, differently to what you may have on supermarkets. In our test cases, the database contained about 50000 items with fields reported in Table I.

TABLE I. PRODUCT ITEM DESCRIPTIONS FIELDS

Field ID	Item Description	Example
TYPE	Type	"1A0145", "1A0333",...
CONFIGURATION	Configuration	"DRESS", "JACKET",...
PATTERN	Color	"White", "Red", "Navy blue",...
MODEL	Alphanumeric code model	"1A0145", "1A0333", ...
PACKAGING_TYPE	Type packaging	"Packaging Basic PE", "Packaging Basic-Contin.", "Women's Packaging A/I",
PRODUCTION_CATEGORY	Production category	"Accessories", "Clothing", "Jeans",
MERCHANDISE_MCR_TYPE	Merchandise type	"Basic, Preview", "Women", "Main Women",
MERCHANDISE TYPOLOGY	Merchandise tipology	"Preview Women SS", "Main Women AI", "Women PE",
MERCHANDISE_MCR_FAMILY	Merchandise family	"Coat", "Bag", "Dress",
MERCHANDISE_GROUP	Merchandise group	"Jewelry", "Dress", "Shirt",
GENDER	Gender	"Accessories Women", "Child", "Women",
BRAND	Brand	"VA", "GM", "PW",
STYLE_GROUP	Style	"P", "C",
BIRTH_SEASON	Season	"20201", "20062", "20071",
PERIODICITY	Periodicity	"C", "S",
IS_CLOTHING_ITEM	Marking if the item belongs to a clothing category	1,0 (yes/no)
NRM_CAT_LVL_1	Code normalized business classification level 1	"Accessories", "Clothing", "Jeans",
NRM_CAT_LVL_2	Code normalized business classification level 2	"Bag", "Clothing", "Coat",
NRM_CAT_LVL_3	Code normalized business classification level 3	"Shopping", "Dress", "Jacket",
NET_SOLD_PRICE	Price	1580.00

Field ID	Item Description	Example
IN_STOCK	Whether an item is available or not	1,0 (yes/no)
132 X Hashtag tasche, abalze,...	Hashtag website	1,0 (yes/no)

Most of the fields are strings, only a few of them are numeric or Boolean. Clustering has been carried out by using K-medoids [20]. K-medoid is a classical clustering technique that partition a dataset of n objects into k a priori known clusters. A number of techniques can be used to identify the best compromise on the value of K [2]. To calculate the distance among items we used the Gower distance [13], which is computed as the average of partial dissimilarities across individuals. Each partial dissimilarity (and thus the Gower distance) ranges in [0,1].

$$d(i,j) = \frac{1}{p} \sum_{i=1}^p d_{ij}^{(f)}$$

Where: $d_{ij}^{(f)}$ is the partial dissimilarity computation which depends on the type of variable being evaluated. For a qualitative assessment the partial dissimilarity is 1 only if observations x_i and x_j have different values, and 0 otherwise. Through the silhouette method we determine the optimal number of clusters. The silhouette method calculates the average silhouette of observations for different values of K [2]. The optimal number of clusters K is the one that maximizes the silhouette over a range of possible values for K. In Figure 3a, the trend of silhouette index, as a function of K, is reported. Figure 3b shows the distribution of cluster size for K=13. From the trend, the value of K=13 has been estimated as a good compromise. In Table II, the descriptions of the identified clusters, and the corresponding sales are reported. The main descriptions have been identified by a cluster analysis.

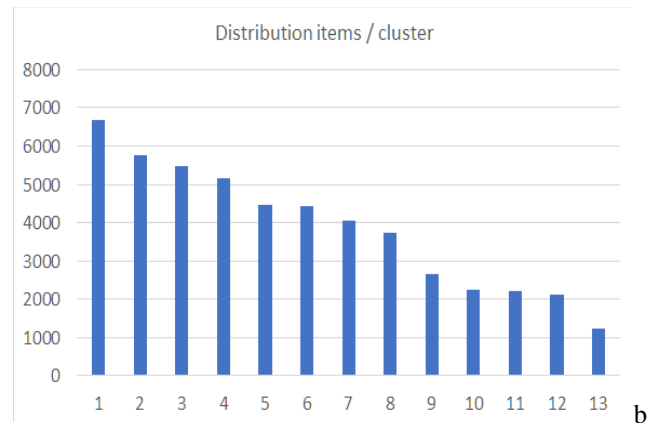
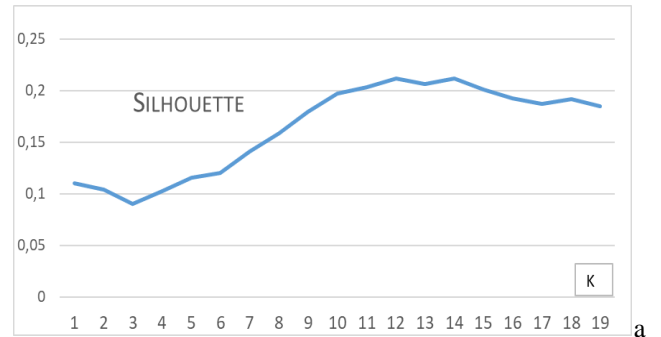


Figure 3 – (a) Trend of the Silhouette value as a function of the number cluster K for item dataset; (b) distribution of items/cluster when K=13.

TABLE II. MAIN DESCRIPTION OF CLUSTERS.

Cluster	Derived descriptions of the item clusters	# sales
1	BAG	969
2	DRESS	1171
3	TROUSERS	794
4	KNIT	678
5	T-SHIRT	674
6	ACCESSORIES (HAT - FOULARD - SCARF - NECKLACE - GLOVES - BRACELET)	596
7	SHIRT	838
8	COAT	388
9	SHOES	341
10	SKIRT	530
11	JACKET	292
12	BELT	237
13	CHILDREN'S CLOTHING	126

B. Features engineering for customers

The data collected by the administrations and the retail shops refer to the user behavior, which is associated with the user profile. The user profile has been enriched with information regarding customer behavior adding: (i) fields about the customer's maximum interest for an item within the cluster, such as: Interest (Yes/no), Observed (Totem, Online, etc.), Tried, Purchased item; (ii) fields describing the items purchased within the cluster.

In addition, a number of features (which in some sense are KPI, Key performance indicators) have been also computed, assessed, identified and discussed in this section by taking into account the experience of business developers. Among them: recency, frequency, and average spending. *Recency* is defined as the number of days passed since the last visit or access in a store or online; *Frequency* represents the frequency of purchase in number of days; *Average spending* is the average value of single ticket for the customer (estimated on the basis of the admin track record). In addition, in order to distinguish from online and in-store behavior, online and in-store frequency and recency have been computed.

C. Second Level Clustering on user profiling

In this case, the number of user profiles has been 608447, of which 27346 have been acquired in the 2016-2019 temporal range. The user profile includes the features listed in Table III.

TABLE III. 35 USER CUSTOMER FEATURES (ALL NUMBERS).

Name profile feature	Description
RFM_TRN_DaysFrequency	Frequency transaction
RFM_TRN_DaysRecency	Recency transaction
RFM_TRN_AvgAmount	Average spending transaction
RFM_PRS_ONLINE_DaysFrequency	Frequency presence online
RFM_PRS_ONLINE_DaysRecency	Recency presence online
RFM_PRS_ONPREM_DaysFrequency	Frequency presence store
RFM_PRS_ONPREM_DaysRecency	Recency presence store
FidelityUsageRange	Fidelity card use
CUS_FIDELITY_CARD_LEVEL_CD	Fidelity card level
Cluster_k_Interest size[13]	Max interest for each cluster
Cluster_k_Purchased size[13]	Number of items purchased

On the basis of the user profile features, which include two vectors of preferences of the user for items clusters identified in the first phase, a clustering has been carried out through the K-means method. The Silhouette method has been used to determine the optimal number of clusters, in this case $k=14$ (see Figure 4). In Table IV, the derived descriptions of customers/user clusters and they corresponding size are reported.

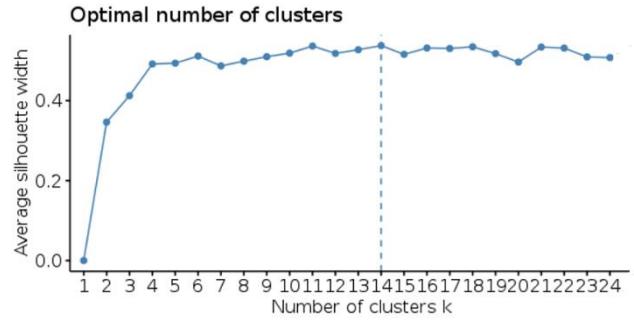


Figure 4 - Average silhouette width vs number of clusters

TABLE IV. DESCRIPTION OF USERS' CLUSTERS.

Cluster	Derived Description from Customer cluster analysis	# total customer
1	Customers with average spending amount not defined; the frequency is not defined neither in store neither online; day of the last purchase not defined	9195
2	Customers with low average spending amount, mainly online with undefined frequency and last purchase older than two years	3158
3	Customers with undefined average spending amount, mainly in store, with undefined frequency and last purchase older than two years mainly online	2433
4	Customers with low average spending amount, last purchase older than one year.	2302
5	Customers with low average spending amount in store, with frequency of about 4 months in store; last purchase has been made within one year. often using the fidelity card	2302
6	Customers with low average spending amount, more frequent in store with annual frequency; last purchase older than one year.	1657
7	Customer with low average spending amount, more frequent online, but also buying in store with frequency of about 2 months online and about 6 months in store; last purchase older than one year, use fidelity card	1493
8	Customer with average spending amount not defined, mainly online; last purchase mid term days	1186
9	Customer with very high average spending amount in store	887
10	Customer with medium average spending amount more frequent in store but also buys in store with frequency about 230 days; last purchase about 262 days, use fidelity card	819
11	Customer with average spending medium amount in store; last purchase one year ago; frequency is not defined	797
12	Customer with average spending amount not defined, mainly online, with frequency of about 270 days; last purchase one year	717
13	Customer with medium average spending amount, mainly in store, with not defined frequency and last purchase older than one year	391
14	Online customers with annual frequency	9

According to the obtained results, cluster #1 was actually very large. For this reason, a second level clustering has been performed to split user cluster #1 in subclusters based on the same features. Also in this case, the silhouette method has been used to identify the best compromise in terms of the number of clusters. Analyzing the distributions of cluster size, we opted for $K=5$, with the aim of having maximum classifications and expression, as shown in Table V. The final distribution of clusters has been reported in Figure 5.

TABLE V. DESCRIPTION OF SECOND LEVEL CLUSTER OF CLUSTER #1.

Cluster	Derived Description from Customer cluster analysis	# total customer
1.1	Customers with average spending amount undefined; the frequency is undefined neither in store nor online; day of the last purchase undefined	5167
1.2	Customers with low average spending amount. They mainly buy in the product cluster #12	2411
1.3	Customers with very low average spending amount, mainly in the product clusters: #2, #10 and #12	1330
1.4	Customers with: recency of about 23 days, frequency of about 18 days	173
1.5	Customers with average spending amount of about 150 Euro; mainly buying in the product cluster #1	148

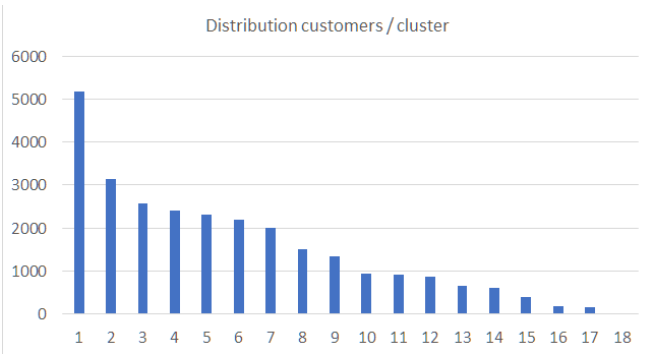


Figure 5 - Distribution of customers along the resulting 18 clusters

D. Final production of suggestions

As described above, the identified solution produces a number of recommendations for each user. Each possible suggestion is labeled with the kind, the date of emission, and a deadline. The Engager Tool also marks those that have been spent with: the date and time of emission, the channel adopted (shopID, mobileApp, website, shopID.totemID, etc.), the ID of the assistant, etc. This information is useful for the assessment of the level of acceptance, and thus for the validation, as described in the next Section. Therefore, the database with the suggestions is never discharged since the recommender has to take into account the already spent suggestions.

The recommendations are generated according to different kinds (as described in the following list) and they are consumed in different contexts by the Engager Tool. The **different kinds of recommendations** are by:

customer similarity: for each customer cluster the most representative items are built. They are identified among the most purchased items within the users' ones belonging to the same item cluster (they can be selected by using other criteria, for example: because they are the most frequently asked, or the company would like to push them, or they are closer to the cluster centroid, or to maximize the revenue or minimize the stock, etc.). In addition, the suggested item should have not been already purchased or proposed/suggested to same customer in the same season.

item similarity: considering the last items purchased by the customer according to the information contained into its profile, and randomly selecting items in the same item clusters, avoiding proposing items which have been already bought or proposed. Also in this case, the items can be

filtered/selected by using additional criteria, for example: because they are the most frequently asked, or the company would like to push them, or they are closer to cluster centroid, or to maximize the revenue or minimize the stock, etc.

item complementary: considering items that may complement the last items that have been bought by the customer according to a table of complementary items; for example: a belt in combo with a bag. Please note that some of the item clusters are complementary each other, see the above descriptions – e.g., #1 and #2 of Table II. To this end, through association rules using a priori algorithm [10] for each transaction in the dataset a set of metrics have been calculated; some examples are reported in Table VI, for the first 5 clusters. The used metrics are *support*, *confidence*, *lift*, *count*, and are defined as follows. Let N and M be two clusters. $Support(\{N\} \rightarrow \{M\})$ is the ratio of the number of transactions/tickets including N and M wrt the total number of transactions. $Confidence(\{N\} \rightarrow \{M\})$ is the ratio of the number of transactions containing N and M wrt the total number of transactions containing N . $Lift(\{N\} \rightarrow \{M\})$ is the ratio of confidence of N wrt the total number of transactions containing M . $Lift(\{N\} \rightarrow \{M\})$ is the number of transactions containing N or M . To generate the recommendations, we considered the top 5 clusters with highest *support* and suggested one of the best-selling items (*count*) within the cluster.

TABLE VI. EXAMPLE OF COMPLEMENTARY CLUSTERS ASSESSMENT BY USING METRICS: SUPPORT, CONFIDENCE, LIFT AND COUNT (PART)

Cluster	Complementary clusters				
	cluster	support	confidence	lift	count
1	2	0.26486066	0.6069351	1.106003	12935
	7	0.24864345	0.5697729	1.253423	12143
	3	0.24465057	0.5606231	1.213722	11948
	8	0.24336057	0.5576670	1.277549	11885
2	4	0.22298667	0.5109797	1.282096	10890
	3	0.34351004	0.6259701	1.355196	16776
	7	0.32391425	0.5902612	1.298495	15819
	8	0.31392182	0.5720522	1.310504	15331
3	4	0.29840080	0.5437687	1.364367	214573
	2	0.34351004	0.7436830	1.355196	16776
	7	0.30397035	0.6580814	1.447690	14845
	8	0.29868747	0.6466442	1.481385	14587
4	4	0.27753548	0.6008511	1.507592	13554
	1	0.24465057	0.5296569	1.213722	11948
	2	0.29840080	0.7487156	1.364367	214573
	3	0.27753548	0.6963625	1.507592	13554
5	7	0.26578209	0.6668722	1.467029	12980
	8	0.27260069	0.6839807	1.566918	13313
	1	0.22298667	0.5594945	1.282096	10890
	2	0.13366914	0.7559931	1.377628	6528
2	8	0.12396339	0.7011002	1.606137	6054
	7	0.12224338	0.6913723	1.520926	5970
	3	0.12199767	0.6899826	1.493780	5958
2	4	0.12158814	0.6876665	1.725420	5938

item associated: in order to improve a customer's purchase frequency, we generated suggestions for customers who purchased an item in the last three months. For the generation we have proceeded as follows: through association rules using a priori algorithm [10] we have defined pairs of items (i, j) with $support \geq 0.001$ and $confidence \geq 0.01$. If a customer buys item i then item j will be suggested. This is the typical suggestion which can be delivered for stimulating the return on the shop.

suggestions for serendipity: randomly selecting items to be suggested from the whole present collection, taking also into account what is available in the physical shop.

V. ASSESSMENT AND VALIDATION

The recommendation system has been validated in a store located in Florence and on the online store as follows. We have exploited the data collected since December 2019 to test and tune the solution, verifying if the suggestions produced were also provided by the Assistant in shops and finally acquired by the customers. The algorithm updates the clusters monthly and generates the new suggestions daily. With the suggestions generated, without stimulating customers, we verified in the period January - June 2020, through transactions and verifying the shop assistants (which are the reference experts), if there was a match between suggestions and items purchased by customers. This analysis showed that on about 400 customers who bought, about 10000 suggestions were generated. On suggestions generated, the 6.36% items were purchased. This was considered the minimum level of reaching with the efficiency since resulted to be possible without the tool. Then from July 2020 until December 2020, the recommendation system was tuned on operative to stimulate a certain class of users, entering in the store, using the totem in the store and by mail for ecommerce. This analysis with the stimulated customers showed that on 67 selected customers in the trial, 3050 suggestions have been generated, while only about the 20% has been actually sent to the customers (on shops and/or email). On the items suggested, the 9.84% of them were actually acquired or tested. Therefore, using the stimulus of the recommendation system, we have increased the customers' attention of the 3.48%. The period for the assessment and validation was also complicated by the COVID-19 pandemic which strongly limited the access to the stores, and the validation via the e-commerce without the effective verification of the shop assistant is not comparable with the conditions of the 2019.

VI. CONCLUSIONS

In this paper, a recommendation system in the context of fashion retail has been proposed and described, relying on a multi-level clustering approach of items and users' profiles in online and physical stores. The solution has been developed in the context of the Feedback project founded by Regione Toscana, and has been conducted on real retail company Tessilform, and it has been validated against real data from December 2019 to December 2020, showing that the use of the proposed recommendation tool generated stimulus to the customers which brought to an increase of buyers' attention and purchase increase of 3.48%. The solutions proposed has demonstrated to be functional also in the presence of low number of customers and items, and when suggestions are mediated by the assistants as happen in the fashion retail shops. Moreover, the proposed solution addresses and solved lacks and issues which are present in current state of the art tools, such as also the cold start problems in generating recommendations for newly acquired customers, since it relies on rules mining techniques, allowing to predict the purchase behavior of new users. Our solution is also GDPR compliant, addressing the current strict policies for users data privacy, solving one of the main issue for managing users' demographic details.

ACKNOWLEDGMENT

The authors would like to thank FEEDBACK project and partners for which we have developed a part of the solutions described in this paper, and Regione Toscana for the partial founding POR FESR 2020 Phase 2. <https://www.vargroup.it/progetti-rd/>

REFERENCES

- [1] Michael, J.A Berry A., Gordon S. Linoff. "Data Mining Techniques Second Edition", wiley publishing.
- [2] U. Maulik and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 12, pp. 1650–1654, 2002.
- [3] M. Namvar, M. R. Gholamian and S. KhakAbi, "A Two Phase Clustering Method for Intelligent Customer Segmentation," 2010 International Conference on Intelligent Systems, Modelling and Simulation, Liverpool, 2010, pp. 215-219.
- [4] Michael Giering. 2008. Retail sales prediction and item recommendations using customer demographics at store level. SIGKDD Explor. Newsl. 10, 2 (December 2008), 84–89. DOI:<https://doi.org/10.1145/1540276.154030>
- [5] Chen, D., Sain, S. & Guo, K. Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining. J Database Mark Cust Strategy Manag 19, 197–208 (2012). <https://doi.org/10.1057/dbm.2012.17>.
- [6] Su-Yeon Kim, Tae-Soo Jung, Eui-Ho Suh, Hyun-Seok Hwang, "Customer segmentation and strategy development based on customer lifetime value: A case study" Expert systems with applications, Vol. 31, pp. 101-107, 2006.
- [7] John A. McCarty, Manoj Hastak, "Segmentation approaches in data-mining: A comparison of RFM, CHAID, and logistic regression", Journal of Business Research, Vol. 60, pp. 656–662, 2007.
- [8] Hyunseok Hwang, Taesoo Jung, Euiho Suh, "An LTV model and customer segmentation based on customer value: a case study on the wireless telecommunication industry" Expert systems with applications, Vol. 26, pp. 181-188, 2004.
- [9] Chu Chai Henry Chan, "Intelligent value-based customer segmentation method for campaign management: A case study of automobile retailer", Expert systems with applications, Vol. 34, pp. 2754-2762, 2008.
- [10] Mining association rules between sets of items in large databases R. Agrawal, T. Imielinski and A. Swami ACM SIGMOD Rec., 22 (2) (1993), pp. 207-216
- [11] Mining frequent patterns without candidate generation Han J., Pei J. and Yin Y. ACM SIGMOD Rec., 29 (2) (2000), pp. 1-12
- [12] SS-FIM: single scan for frequent itemsets mining in transactional databases Y. Djenouri, M. Comuzzi and D. Djenouri Proceedings of the 21st Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer (2017), pp. 644-654
- [13] J. C. Gower, "A general coefficient of similarity and some of its properties," Biometrics, pp. 857–871, 1971.
- [14] Fátima Rodrigues, Bruno Ferreira "Product Recommendation based on Shared Customer's Behaviour" <https://doi.org/10.1016/j.procs.2016.09.133>
- [15] Tuinhof H., Pirker C., Haltmeier M. (2019) Image-Based Fashion Product Recommendation with Deep Learning. In: Nicosia G., Pardalos P., Giuffrida G., Umeton R., Sciacca V. (eds) Machine Learning, Optimization, and Data Science. LOD 2018. Lecture Notes in Computer Science, vol 11331. Springer, Cham
- [16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, pages 1097–1105, 2012.
- [17] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, pages 448–456, 2015.
- [18] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In Proceedings of the 26th International Conference on World Wide Web (WWW '17). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 173–182. DOI:<https://doi.org/10.1145/3038912.3052569>

- [19] R. Mu, "A Survey of Recommender Systems Based on Deep Learning," in *IEEE Access*, vol. 6, pp. 69009-69022, 2018, doi: 10.1109/ACCESS.2018.2880197
- [20] Kaufman, L., & Rousseeuw, P. J. (1990). *Finding groups in data: An introduction to cluster analysis*. New York: Wiley.
- [21] S. Wu, W. Ren, C. Yu, G. Chen, D. Zhang and J. Zhu, "Personal recommendation using deep recurrent neural networks in NetEase," 2016 IEEE 32nd International Conference on Data Engineering (ICDE), Helsinki, 2016, pp. 1218-1229.
- [22] C. Badii, P. Bellini, A. Difino, P. Nesi, "Smart City IoT Platform Respecting GDPR Privacy and Security Aspects", *IEEE Access*, 2020. 10.1109/ACCESS.2020.2968741 <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&number=8966344>
- [23] Nageswara, R. K. , & Talwar, V. G. (2008). Application domain and functional classification of recommender systems a survey. *DESIDOC Journal of Library and Information Technology*, 28 (3), 17–35 .
- [24] C. Wang, M. Niepert and H. Li, "RecSys-DAN: Discriminative Adversarial Networks for Cross-Domain Recommender Systems," in *IEEE Transactions on Neural Networks and Learning Systems*.
- [25] M. Ho&ma and A. Swami. Set-oriented mining of association rules. Research Report RJ 9567, IBM Almaden Research Center, San Jose, Cali- fornia, October 1993.
- [26] Cho Y.H., Kim J.K., Kim S.H. A personalized recommender system based on web usage mining and decision tree induction *Expert Syst. Appl.*, 23 (2002), pp. 329-342 International Conference on Innovative Internet Community Systems (I2CS) – Jubilee Edition 2010 –. Bonn: Gesellschaft für Informatik e.V.. (S. 170-179).
- [27] Pazzani, M., Billsus, D. Learning and Revising User Profiles: The Identification of Interesting Web Sites. *Machine Learning* 27, 313–331 (1997). <https://doi.org/10.1023/A:1007369909943>.
- [28] Billsus, D., Pazzani, M.J. User Modeling for Adaptive News Access. *User Modeling and User-Adapted Interaction* 10, 147–180 (2000). <https://doi.org/10.1023/A:1026501525781>
- [29] Tariq Mahmood and Francesco Ricci. 2009. Improving recommender systems with adaptive conversational strategies. In *Proceedings of the 20th ACM conference on Hypertext and hypermedia (HT '09)*. Association for Computing Machinery, New York, NY, USA, 73–82. DOI:<https://doi.org/10.1145/1557914.1557930>.
- [30] Min SH., Han I. (2005) Recommender Systems Using Support Vector Machines. In: Lowe D., Gaedke M. (eds) *Web Engineering. ICWE 2005. Lecture Notes in Computer Science*, vol 3579. Springer, Berlin, Heidelberg
- [31] Ricci F., Rokach L., Shapira B. (2011) *Introduction to Recommender Systems Handbook*. In: Ricci F., Rokach L., Shapira B., Kantor P. (eds) *Recommender Systems Handbook*. Springer, Boston, MA.
- [32] Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. 2015. AutoRec: Autoencoders Meet Collaborative Filtering. In *Proceedings of the 24th International Conference on World Wide Web (WWW '15 Companion)*. Association for Computing Machinery, New York, NY, USA, 111–112. DOI:<https://doi.org/10.1145/2740908.2742726>
- [33] S. Wei, N. Ye, S. Zhang, X. Huang and J. Zhu, "Collaborative Filtering Recommendation Algorithm Based on Item Clustering and Global Similarity," 2012 Fifth International Conference on Business Intelligence and Financial Engineering, Lanzhou, 2012, pp. 69-72
- [34] Xiangyu Zhao, Liang Zhang, Zhuoye Ding, Dawei Yin, Yihong Zhao, and Jiliang Tang. 2017. Deep Reinforcement Learning for List-wise Recommendations. *arXiv preprint arXiv:1801.00209* (2017)
- [35] Y. Koren, R. Bell and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," in *Computer*, vol. 42, no. 8, pp. 30-37, Aug. 2009
- [36] Ngai, E., Xiu, L., & Chau, D. (2009). Application of data mining techniques in cus- tomer relationship management: A literature review and classification. *Expert Systems with Applications*, 36 (2), 2592–2602. <https://doi.org/10.1016/j.eswa.2008.02.021> .
- [37] Swift, R.S., 2001. *Accelerating Customer Relationship using CRM and Relationship Technologies*. Upper Saddle River: Prentice Hall PTR.
- [38] P. Bellini, I. Bruno, D. Cenni, A. Fuzier, P. Nesi, M. Paolucci, "Mobile Medicine: Semantic Computing Management for Health Care Applications on Desktop and Mobile Devices", on *Multimedia Tools and Applications*, Springer. <http://www.springerlink.com/content/q8512555u0j00584/> new May 2012, Volume 58, Issue 1, pp 41-79 (see more)
- [39] P. Bellini, I. Bruno, P. Nesi, M. Paolucci, "A Static and Dynamic Recommendations System for Best Practice Networks", *Proc. of the Human-Computer Interaction Conference*, 21-26- July 2013, Nevada, USA, Press by Springer Verlag LNCS.new
- [40] P. Bellini, D. Cenni, P. Nesi, "Optimization of Information Retrieval for Cross Media contents in a Best Practice Network", *International Journal Multimedia Information Retrieval*, 10-04-2014. <http://link.springer.com/article/10.1007/s13735-014-0058-8?>
- [41] Cuzzocrea, A., Fadda, E., Data-Intensive Object-Oriented Adaptive Web Systems: Implementing and Experimenting the OO-XAHM Framework, *Proceedings of the 12th International Conference on Management of Digital EcoSystems, MEDES 2020*, 2020, pp. 115–123