A Smart Financial Advisory System exploiting Case-Based Reasoning.

Giorgio Leonardi[1] and **Luigi Portinale**[1]
and **Paolo Artusio**[2] and **Marco Valsania**[2]

Abstract. In the financial advisory context, knowledge-based recommendations based on Case-Based Reasoning are an emerging trend. They usually exploit knowledge about past experiences and about the characterization of both customers and financial products. In the present paper, we report the experience related to the development of a case-based recommendation module in a project called SMARTFASI. We present a solution aimed at personalizing the asset picking phase, by taking into consideration choices made by customers who have a financial and personal data profile "similar" to the current one. We discuss the notion of distance-based similarity adopted in our system and how to actually implement an asset recommendation strategy integrated with the other software modules of SMARTFASI. We finally discuss the impact such a strategy may have both from the point of view of private investors and professional users.

1 Introduction

The evolution of the international financial context (often dictated by the worldwide economic and financial crisis) has progressively changed, often in a radical way, the attitude of investors. One direct consequence is that single investors are no longer simply classifiable into as *private*, *retail* and *affluent* in the traditional way; on the contrary, a common aspect among all the different types of investors is the need to have more clarity on the financial products and the possible benefits from tailor-made services. Likewise, there is a change in commercial strategies, switching from different approaches to each market segment. toward the adoption of common strategies covering multiple segments [6]. This can lead to a global standardization of banking services through the identification of common needs among different market segments [13].)

A partial answer to the first issue (i.e., the difficulty in exploiting a traditional investor's classification scheme) has been provided by the introduction of specific norms (as for example the EU MiFID guideline [1]). On the other hand, concerning the needs of the users (i.e., the investors), a rapid evolution of the financial advisory process is taking place; the goal is to provide the user with a financial proposal that is most suitable for the users needs and profile, and goes beyond the consideration of legal issues as the only guideline. For this reason, recommendation strategies are becoming quite popular in the financial advisory context, with particular attention to the Case-Based Reasoning (CBR) paradigm [10, 16, 14, 15]. In general, we can use three main approaches in recommendation operations [12]:

- Collaborative Filtering: assuming that human preferences are correlated, we can collect preferences of a large set of customers in order to define a recommendation based on preferences of people with similar interests.
- Content-based filtering: use of preferences of a specific customer to infer recommendations, based on specific categories (keywords) connected to a profile.
- Knowledge-based: recommendations are based on different levels of knowledge about the product domain.

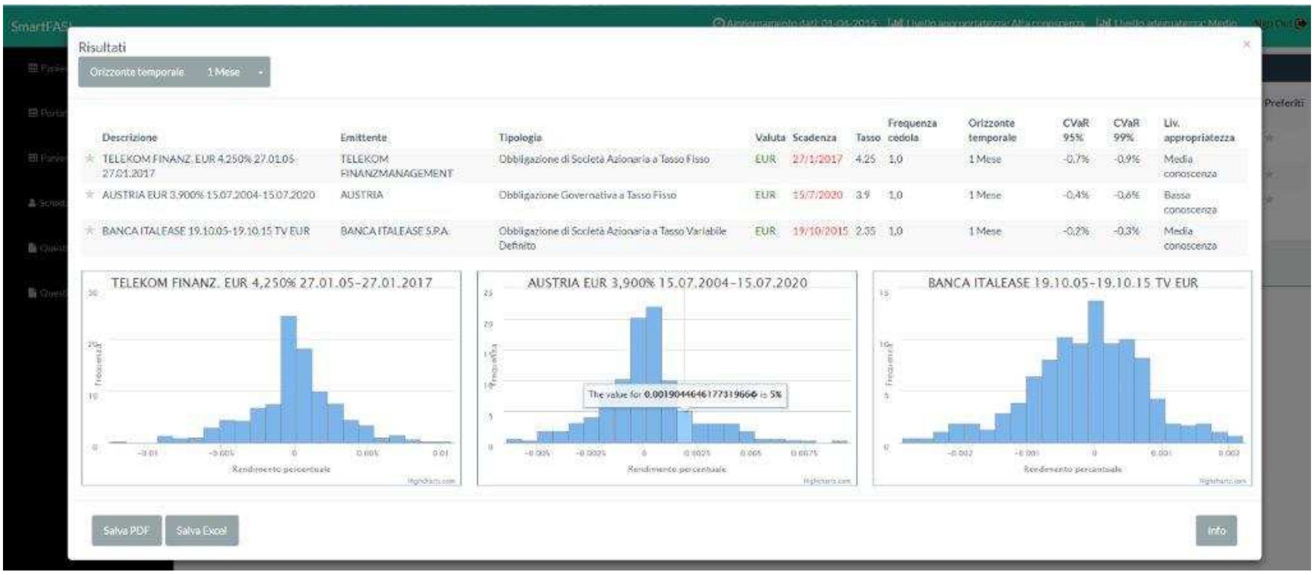
In finance, knowledge-based method (among which CBR) is mostly applied, as investment recommendations must primarily conform to legal regulations (i.e. MiFID) in order to ensure investors against mismatching and/or fraudulent financial proposals [20]. Moreover, historical data are available, making possible, as predicated by the CBR paradigm, to exploit knowledge about past experiences and about the characterization of both customers and financial products.

In addition, thanks to the IT advances, an emerging trend is to base financial services on web and mobile technologies, with strict collaboration between the end-user and the consultant, in such a way as to get the users more and more involved in the final definition of their stock portfolio. In this context, a phase of basic importance is that of *asset picking*; in this phase, advanced data analytic tools are adopted, in order to compare the risk and performance of the considered financial products, perhaps prior filtering of the assets by means of specific features, either identity-based (as asset class, country, region, currency) or measured (as duration, historical volatility, time to maturity, historical performance, etc…). In this paper, we present the solution adopted in the SMARTFASI project, which has the goal of designing and implementing a web-based architecture for a financial decision support system able to supply a set of advanced consultancy services for the management of financial assets, whilst taking into account the risk/performance trade-off. The advisory system prototype has been designed with different goals in mind:

- the exploitation of Cloud and High-Performance Computing (HPC) paradigms at the *infrastructure level*;
- the exploitation of stochastic modeling and Montecarlo simulation, together with Case-Based Reasoning (CBR) [2] at the *methodological level*.

Cloud and HPC infrastructure have been introduced to support stochastic simulation which is a computationally intensive activity. The aim is to provide the user with a set of simulation tools, in such a way that he/she can simulate the assets behaviour in a specific time horizon, by computing for instance the expected yield and indices like the CVaR (Conditional Value at Risk) with a given confidence

[1] Università del Piemonte Orientale, Italy, email:{giorgio.leonardi, luigi.portinale}@uniupo.it
[2] ORS srl, Italy, email:{paolo.artusio, marco.valsania}@ors.it



Risultati

Descrizione	Emittente	Tipologia	Valuta	Scadenza	Tasso	Frequenza cedola	Orizzonte temporale	CVaR 95%	CVaR 99%	Liv. appropriatezza
TELEKOM FINANZ. EUR 4.250% 27.01.05-27.01.2017	TELEKOM FINANZMANAGEMENT	Obbligazione di Società Azionaria a Tasso Fisso	EUR	27/1/2017	4.25	1,0	1 Mese	-0,7%	-0,9%	Media conoscenza
AUSTRIA EUR 3.900% 15.07.2004-15.07.2020	AUSTRIA	Obbligazione Governativa a Tasso Fisso	EUR	15/7/2020	3.9	1,0	1 Mese	-0,4%	-0,6%	Bassa conoscenza
BANCA ITALEASE 19.10.05-19.10.15 TV EUR	BANCA ITALEASE S.P.A.	Obbligazione di Società Azionaria a Tasso Variabile Definito	EUR	19/10/2015	2.35	1,0	1 Mese	-0,2%	-0,3%	Media conoscenza

Figure 1. Comparison of different products

level (e.g., 95% and 99%). This can be done by considering either a single product or by comparing several options. Figure 1 shows an example of products comparison exploiting Monte Carlo simulation [8, 5].

However, the use of simulation tools leave the user alone in the choice of the financial products; for this reason the system has been enriched with a *case-based recommendation engine*, implementing a knowledge-based recommendation strategy, and able to suggest to the users a set of options tailored to their needs. The asset picking phase has then been expanded by taking into consideration, among the others, the frequency of use of products selected by customers who have a financial and personal data profile "similar" to the current one. The underlying assumption is that individuals who share several features (in terms of financial needs), will act on the market in a similar way.

The focus of the present paper is on such a case-based recommendation engine; in the following sections we will details both the methodological issues as well as the architecture on which this part of the SMARTFASI system is based. The exploitation of CBR techniques allows us to address the following targets with potential different end-users:

- **Private Investors**:
  - to improve the vision of the global investment scenario, by putting more emphasis and focus on the individual user features (e.g., financial attitude), producing more informed choices for the users;
- **Professional Users** (e.g., consulting agents or firms):
  - to propose to the customers some investment scenarios which are no more generically based on the financial feastures of the products only, but also more tailored to the specific customer profile, by personalizing in this way the service (for example by comparing benchmarks more suitable to the customers);
  - to exploit new analytical tools to evaluate the value of the set of potential investments, or alternatively to suitably modify this set, in order to fulfil the customers needs, preferences and requirements.
  - to perform historical analyses on clusters of clients, discovering potential trends of investments that may be consequently supported or contrasted, by evaluating the commercial offer in a more informed way
  - to improve customers acquisition process, tying business targets to the interests of the consumers, so boosting the value of the company's clients portfolio

The remainder of the paper will be organized as follows: Section 2 introduces the basics of the CBR paradigm exploited in the recommendation engine, Section 3 discusses the case-based recommendation methodology introduced in the SMARTFASI project, while in Section 4 the basic architecture of the advisory system is outlined. Final considerations are then reported in Section 5.

2 The CBR paradigm

Case-Based Reasoning (CBR) [18] is a problem solving methodology that addresses the task of solving a new problem (the *target case*), by retrieving, and possibly adapting, the solutions of past problems similar to the one to be solved. The basic idea is to store a set of solved cases in a *case library*, and then to re-use such cases when a new problem has to be solved. The main assumption underlying the CBR process is that similar problems have similar solutions; in this way the solution of a past case can be used to address the solution of a new similar case.

CBR is also considered as a *lazy learning* technique [3], in contrast with *eager learning* where a suitable model is constructed from training cases, which are then no longer needed for problem solving. In CBR, training instances are kept in memory and are directly used when a new case is presented as a target. Following the classical framework described in [2], there are four main step in a CBR problem solving session, the so-called 4R's (see figure 2):

- **Retrieve.** It determines the cases that are most similar to the new problem. The notion of similarity is implemented by defining a notion of distance among the case features, and by finally combining such local distances (at the feature level) into a global measure (at

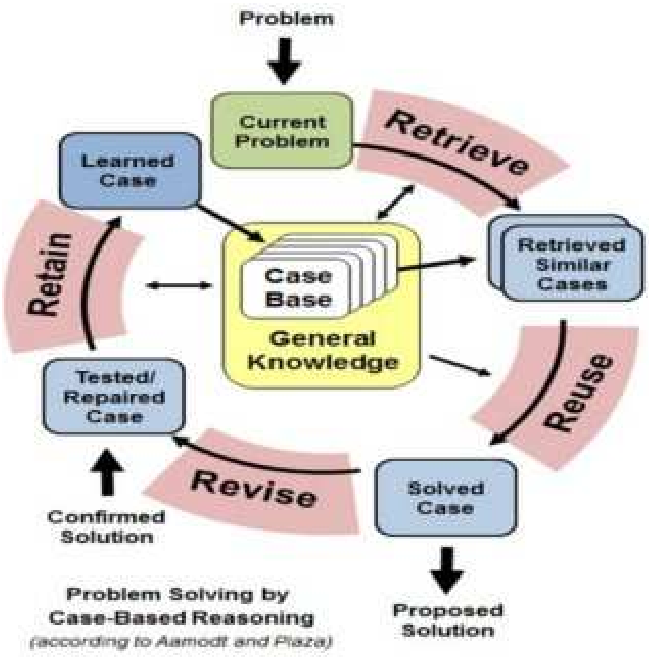


Figure 2. The 4R's CBR cycle

the case level). The retrieve step is usually implemented through k-Nearest Neighbour (kNN) search [19].

- **Revise.** The solution of a retrieved case is selected and proposed as a candidate solution to the new problem. If it can be suitably applied to the target case it becomes a solution for the latter as well. Otherwise, it is passed to the next CBR step.
- **Revise.** This step adapts the candidate solution to the target case, in such a way that it can be applied to it. Knowledge intensive methods can be necessary in this step to perform such an adaptation. If revision is not possible, the system fails in finding a suitable solution to the target.
- **Retain.** This is the actual learning step: it evaluates the obtained solution and it decides whether to retain the new solved case in memory. Because of the well-known *utility problem* [7], not every solution should be stored in the case library, and the case library should be properly maintained (see [17]).

The step that has received most attention is definitely the *retrieve* step; indeed, case retrieval is essential to every application of case-based systems, and in particular to case-based recommendation [12, 4]. Case-based recommendation is usually considered a particular instance of content-based recommendation [11], where cases are typically used to model items, through a classical feature-based description. However, case-based recommenders are more suitably considered as knowledge-based recommenders [12], since they exploit both similarity-based retrieval and general knowledge about users and items (e.g., user's preferences). In fact, one can regard case-based recommenders as collaborative filtering recommenders as well, since the suggestion of similar items to similar users is in principle possible. Instead of directly manipulating matrices of rankings as in standard collaborative filtering approaches [9], they can adopt content-based similarity measures to compare users and their preferences with respect to the items of interest. Next section will discuss the CBR methodology we have introduced in the SmartFasi advisory system, by presenting the details of an asset retrieval strategy based on customer's similarity.

3 Case-Based Recommendation

The SmartFasi recommendation module uses the information available about the customer currently under study (the new or current or target case following the scheme of Section 2), to provide a recommendation of financial products (i.e., the solution in the CBR framework), based on the investments made by similar customers. In this project, the customers are defined by the features presented in Table 1, in Section 3.1. According to the CBR paradigm, similarity is implemented as a proper dual notion of a distance measure. In our case, the distance functions involve features concerning the personal information of the customers, their spending power, their knowledge of the financial domain and the composition of the portfolios they may manage at the moment.

The recommendation strategy takes place as a multi-step procedure. The first step (*Step 1*) performs a selection of the most similar customers with respect to the target one, on the basis of personal data and of the overall composition of their portfolios, as described in detail in Section 3.2. While the above step focuses on the general characteristics of the target customer to retrieve the most similar ones, the next step (*Step 2*, see Section 3.3) concentrates on the investment strategies of these customers, in order to perform a further selection which identifies the subset of the most similar portfolios owned by the previously selected customers, with respect to the portfolios owned by the target one. This means that the recommender module will specifically focus on the financial features only after the first filtering step, thus working on a restricted set of customers who share the same personal data, lifestyle and investment capabilities with the target customer. Moreover, *Step 2* is optional, since its execution depends on whether the target customer already has an active portfolio at the current time. If no active portfolio is available for the target customer, then *Step 2* is not performed, since no portfolio comparison can be made. The third step (*Step 3*, described in Section 3.4), finally extracts the K products to be returned as recommended to the users for their evaluation.

In the rest of this section, we will detail each step described so far, together with the characterization of the features defining a customer and with the distance metrics introduced for the similarity evaluation.

3.1 Case Definition

In the approach we propose, a case describes the characteristics of a customer (the investor) in the SmartFasi system. The customer's features describe their personal characteristics, their investment capablities, their financial adequacy (knowledge of the financial domain) and the composition of any portfolio they hold. As usual in the CBR setting, each of these features is associated with a weight that defines its *importance* (we assume three possible levels of importance: 3 = high, 2 = medium, 1 = low). The features defining a customer and their relative weights are determined by the domain experts involved in the SmartFasi project, and are listed in Table 1.

These features are a mix of heterogeneous information, such as numeric values (Age, Available capital, Adequacy and N. of children), coded information (Marital status, Education, Sex and Type of employment) and arrays (Asset allocation for each portfolio). Among such features, it is worth noting that the Adequacy is a pre-computed value, identifying the ability of the customer to understand the implications of buying financial products having different risk levels. The Adequacy is directly linked with the MiFID profile (see [1]) assigned

Feature #	Feature	Weight
1	Age	3
2	Available capital	3
3	Risk tolerance / Adequacy	3
4	Asset allocation for each portfolio	3
5	N. of children	2
6	Marital status	2
7	Education	2
8	Sex	1
9	Type of employment	1

Table 1. Features defining the personal data of a case

to the customer by the financial organisation which manages his/her interests.

Furthermore, the arrays describing the asset allocation in a case are defined at two different levels, as shown in Figure 3.

In this representation, each single asset can be classified as C (*Corporate*) or G (*Government*). Moreover, each asset can be associated with a *Fixed* (F), *Variable* (V) or *Floating with Cap* (C) rate. The combination of these two classifications generates six different groups of assets, in such a way that each asset stored in the reference data base belongs to one of these groups. For each portfolio, the first-level representation is an array containing the percentage of assets in each of the above 6 classes. Considering all the portfolios of a customer at the first level, it is possible to characterise the general investment preferences of a customer's investment. Since this level of abstraction is useful for characterising the overall investment behaviour, it is exploited together with the other personal data to compose, in the *Step 1* of the recommendation module, the ranking of the customers who are globally more similar to the target one.

The second level representation of the portfolios is an array as well, where each location identifies a specific asset. The contents of the array indicates, for each title, its share in the composition of the portfolio. The description of the portfolios at this level of detail shows which investments have been made by a customer at the maximum granularity available. This information describes exactly the financial behaviour of a customer, therefore it is used, in *Step 2*, in order to select the most similar portfolios, by taking exclusively into account the financial aspects of customers sharing their anagraphical and life-style information with the target one.

In the next subsections, we will detail each specific step on which the recommendation strategy is based.

3.2 Step 1

The first step is devoted to the selection of the most similar customers with respect to the query one, using the personal information shown in Table 1; this focus on the general characteristics of the target customer, without taking into account the financial preferences yet. This selection is performed through a Nearest Neighbour search [22], comparing the query with the cases stored in the case library and cutting the results to the first N best matches. The value of N can be set by the system as a default value (for example, a given percentage of the number of cases in the case base), or provided by the user while defining the query. Since the cases are composed by features of different types, the Heterogeneous Euclidean-Overlap Metric (HEOM) is a natural choice for distance defintion [21]. Consider a given feature f with possible values $x, y \in \texttt{range}(f)$, the HEOM metric is defined as follows:

$$D^{f}_{HEOM}(x,y) = \begin{cases} 1 & \textit{if x or y is unknown} \\ overlap(x,y) & \textit{if f is nominal} \\ rn\_diff(x,y) & \textit{otherwise} \end{cases} \qquad (1)$$

The first possibility of Eq. 1 refers to the situation where the feature f has no value either in the target or in the retrieved case (or in both). In case of a nominal feature, $overlap$ is an $n \times n$ square matrix ($n = |\texttt{ range (} f \texttt{) }|$), where $overlap(x,y) \in [0,1]$ measures the distance between values x and y of f (in the extreme case $overlap(x,y) = 0$ if $x = y$ and $overlap(x,y) = 1$ if $x \neq y$).

Finally, $rn\_diff(x,y) = \frac{|x-y|}{\texttt{range}(f)}$ is the range normalized absolute difference of the feature values, in case of a linear (e.g. numeric) feature. The range of each linear feature f is updated every time a new case is added to the case base, in order to keep the rn_diff in the $[0,1]$ range for each linear feature, preserving the retrieval order of the customers. The definition in Eq 1, has the advantage of returning a distance value in the range $[0,1]$; similarity can then be expressed as $S^f(x,y) = 1 - D^{f}_{HEOM}(x,y)$ where $S^f(x,y) = 1$ means perfect similarity and $S^f(x,y) = 0$ means total dissimilarity.

By considering Table 1, features $1, 2, 3, 5$ and 8 are treated as linear features. On the other hand features $6, 7, 8$ and 9 are considered nominal and an appropriate distance matrix is adopted for each feature. What cannot be dealt with by the standard HEOM metric is the portfolio representation in a case. However, in *Step 1* we need to compare also first-level portofolios among cases. Since this information is stored as an array, a natural choice is to consider a local metric based on *cosine distance*; this choice is well justified in the financial domain where it has been adopted in several advisory systems [10, 15].

Given two arrays $\mathbf{a} = (a_1, a_2, ..., a_n)$ and $\mathbf{b} = (b_1, b_2, ..., b_n)$, the cosine distance between $\mathbf{a}$ and $\mathbf{b}$ is defined as:

$$D_{cos}(\mathbf{a}, \mathbf{b}) = 1 - \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}} \qquad (2)$$

Since in our application every component of the array is non-negative, the above definition returns a value in the range $[0,1]$. In particular, the asset allocation contained in a case is composed by a set of different portfolios, each one represented as a two-level array (as shown in Figure 3). The goal in comparing asset allocations is to determine the best match between the portfolios associated with the retrieved case and the portfolios owned by the target customer.

The strategy implemented in SMARTFASI is the following. Let $P_t = (p_t^1, p_t^2, \ldots, p_t^n)$ and $P_c = (p_c^1, p_c^2, \ldots, p_c^m)$ be the set of first-level portfolio arrays owned by the target customer t and a given customer c respectively (customer c is the one we are comparing the target to); each p_t^i and p_c^j are then arrays corresponding to the first-level representation of a specific portfolio for user t and c respectively.

Let $Perm(P)$ be a permutation of a set of portfolios P; the best match is the pair of permutations $\langle\ P'_t,\ P'_c\ \rangle \in Perm(P_t) \times Perm(P_c)$ resulting in the minimum overall distance D_p between the portfolios as defined in Eq. 3.

$$D_p(P_c, P_t) = \min_{Perm(P_c) \times Perm(P_t)} \frac{\sum_{i=1}^{min(n,m)} D_{cos}\left(p_c^i, p_t^i\right)}{min(n,m)} \qquad (3)$$

The best matching portfolios of user t and c are then extracted as shown in Eq. 4.

$$(P'_t, P'_c) = \underset{Perm(P_c) \times Perm(P_t)}{\arg\min} D_p(P_c, P_t) \qquad (4)$$

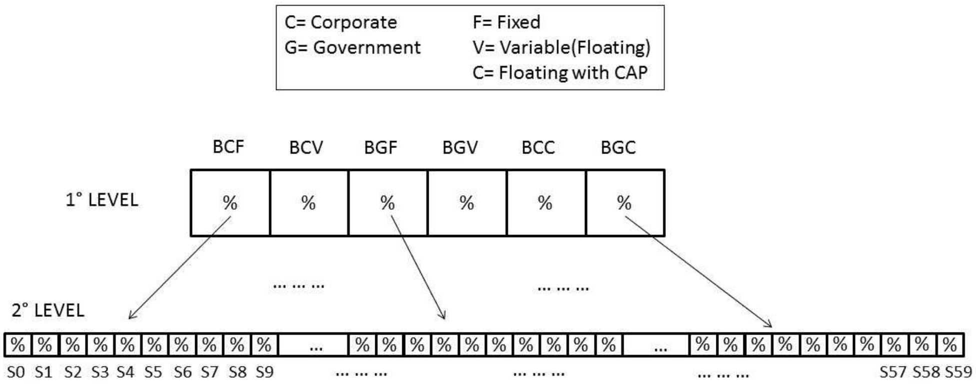


Figure 3. Two-level description of a portfolio

In particular, if one customer (either t or c) has more portfolios than the other, then the portfolios in excess in any given permutation are discarded. Since we consider any possible permutation, they are taken into account when a different permutation is considered. The best matching portfolios for each customer c (i.e., P_c' in Eq. 4) are finally stored in order to be re-used in *Step 2*. In case the target customer has no available portfolio yet, then we consider the asset allocation as a missing feature and we set $D_p\left(P_c, P_t\right) = 1$ as in the HEOM metric.

Finally, once the local distance for each feature has been computed (including the portfolio's distance), the overall distance function between two customers C_1 and C_2 is the normalized weighted average of all the local contributions:

$$D\left(C_1, C_2\right) = \frac{\sum_{i=1}^{s} w_i \cdot D\left(v_1^i, v2^i\right)}{\sum_{i=1}^{s} w_i} \qquad (5)$$

where s is the number of features ($s = 9$ in our application as shown in Table 1), v_1^i, v_2^i are the values of the i-th feature of customer C_1 and C_2 respectively, and w_i the importance weight of the i-th feature (see Table 1 again); furthermore

$$D(v_1^i, v2^i) = \begin{cases} D_p(v_1^i, v_2^i) & \textit{if } i = 4 \textit{ in } \text{Table 1} \\ D^i_{HEOM}(v_1^i, v_2^i) & \textit{otherwise} \end{cases}$$

The global distance defined by Eq. 5 is applied to compare the target customer with all the customers in the case base, in order to obtain the list of the N most similar customers to the target one. This list is then input to *Step 2* if the target customer owns at least one portfolio, in order to further filter these results using the financial information available; otherwise the list is a direct input to *Step 3* since *Step 2* is not applicable.

3.3 Step 2

In *Step 2* the system receives from *Step 1* the list of the N customers globally more similar to the target one, together with the list of the portfolios that best match the portfolios of the target customer. The set of such best matching portfolios is considered and a further filter over the financial information is applied; the goal is to extract the best assets to be recommended, by considering the specific allocations (second-level portfolio information) of such pre-selected similar customers.

Technically, the cosine distance over the arrays representing the second-level description of a portfolio is applied; this level of description details the percentage of investment of each individual asset, while the first-level description (exploited in *Step 1*) details only the percentage of the general classes of investment to which the individual assets belong. In this step, we then concentrate our attention on the actual behaviour of the considered investors, comparing their investment strategies asset by asset.

The output of this phase is a ranked list of portfolios, extracted from the most similar users. An optional system parameter can then be set to cut such a list to the J most similar portfolios, if they are more than J. The aim is to provide the next phase (*Step 3*) with a set of interesting assets, extracted from the most similar portofolios of the most similar customers.

3.4 Step 3

Step 3 receives as input either the ranked list of the J most similar portfolios selected at *Step 2*, or the list of the N most similar customers selected at *Step 1*, if *Step 2* was not applicable. In the latter case, every portfolio belonging to the N most similar customers is extracted, and ranked by user similarity; this means that in both cases this phase consider a ranked list of portfolios (i.e., asset allocations) as input. Starting from this list of asset allocations, the system derives the assets to be returned to the user. This is simply done by looking at the individual assets contained in the list of portfolios, by possibly limiting the set of assets to the first K products found by examining the portfolios in the order provided by their ranking.

In order to provide a more informed decision support, each asset is further associated with some statistics; they can help the user to analyze the provided recommendation, by evaluating a broader spectrum of information. These values are summarized below:

1. *Frequency (F)*: it is the frequency of the asset in the set of retrieved portfolios. For example, if the asset is part of 2 retrieved portfolios out of 5 (i.e. the list input to *Step 3* contains 5 portfolios), then $F = 0.4$.
2. *Average Percentage (AP)*: it represents the average percentage of the considered asset with respect to the retrieved portfolios where it appears. For example, if the asset is part of 2 retrieved portfolios and has a 30% allocation in portfolio p_1 and a 50% allocation in portfolio p_2, then $AP = 40\%$.

3. *Average Distance of Customers (ADC)*: it summarizes the average (global) distance of the retrieved customers who possess the considered asset, using the distance metric described in Eq 5. For example, if the asset is part of the portfolios of 3 customers C_1, C_2, C_3 who are retrieved as similar to the target one in *Step 1*, then the distance between each pair is computed using Eq 5 and then averaged (i.e., $ADP = \frac{D(C_1,C_2)+D(C_2,C_3)+D(C_1,C_3)}{3}$).
4. *Average Distance of Portfolios (ADP)*: it summarizes the average distance of the retrieved portfolios containing the asset (the computation is clearly similar to that of ADC). If the target customer does not have any portfolios, this value is not calculated.

In particular, the F statistic is considered particularly useful, since both most frequently and less frequently used assets (among those recommended) are usually interesting for several reasons. In fact, if the user is a private investor, it could be interesting to him/her to consider which are the financial products that are most popular among users similar to him/her; on the other hand, if the user is a professonal one (e.g., a consultant), then it could be important to analyze the set of products that are not yet popular among the ones that can be recommended to the customers, since it could be a way of differentianting the offer. Moreover, differently from other recommendation situations, in the SMARTFASI context, it makes sense to consider, in the recommended list, also products already owned by the customer, since this may be food for toughts. For example, a private investor can receive confirmation from the fact that an asset present in one of his/her portfolios is pretty popular among similar customers, and he/she may decide to increase the percentage of such an asset; or he/she can discover that one of his/her assets is not very popular among similar customer, and to decide to reduce the percentage in the corresponding portfolio. In any case, finding among the recommended financial products some of their assets can trigger interesting analyses form the customer point of view (either if perfomed directly by the customer in case of a private investor, or if performed by a consultant for the customer's benefit).

Finally, before presenting the user with the list of recommended assets, the system removes those assets which are not compliant with the level of financial knowledge of the target customer; in this way, the system avoids recommending financial products which are not compatible with the customer's MiFID profile. This is done by comparing the risk level of each product with the level of the user's financial adequacy (feature 3 in Table 1).

The final list of products is then presented to the user who can then inspect each asset, by visualizing together with the associated statistics mentioned above, all the basic characteristics of the financial product, as well as its perfomances, both historical and simulated. In the current version of SMARTFASI, such a list is also ordered by frequency F.

4 System Architecture

In this section we discuss the implementation of the recommendation subsystem of the SMARTFASI project. The general architecture of the recommendation module and its integration/interaction with the other parts of the SMARTFASI software is illustrated in Figure 4. In fact, the SMARTFASI advisory system is a web-based application following a standard 3-tier architecture as follows:

- a *web/mobile browser* providing the client level and user interface,
- an *application server* organized into several submodules
  - a *middleware* receiving requests from the client and dispatching them to the requested service manager
  - a *simulation engine*, providing the Monte Carlo simulation service
  - a *recommendation module*, providing the recommendation service which is the focus of the present paper
- a client/server *RDBMS*, providing the data tier where information about customers and financial products are stored.

The recommendation module (Recommender subsystem, in Figure 4) is implemented in JAVA as a standard TCP server; even if part of the whole application server of SMARTFASI, the recommender subsystem can in principle be separated from it, resulting in an independent module that can be remotely queried from multiple installations of the SMARTFASI middleware. Indeed, the middleware acts as a client of the recommendation module through a standard client-server interaction and communication.

Concerning a recommendation session, at the browser level, the software interacts with the user whose requests are sent to the middleware; the latter then builds one or more queries, containing both the target customer(s) identification code(s) and all the requested query parameters. These queries are then sent to the recommendation module through a TCP request message. The recommendation module, on the other side, acts as a server, so it is constantly waiting for requests from the middleware. For each submitted query, the server checks its syntax and, in case of positive response, creates a new instance of the recommendation engine, which performs all the steps described in Section 3. Each instance is encapsulated in a new thread, created by the recommender subsystem to handle each query separately. This mechanism creates a robust and responsive server, able to properly act even if one or more instances of the recommendation module unexpectedly fail. It is also able to effectively distribute the workload when many queries must be satisfied simultaneously. Every time an instance terminates its computation, it communicates the query results to the middleware through a TCP answer. If no answer reaches the middleware within a maximum time limit (due to any unexpected error occurred to the relative server instance), the middleware module closes the TCP connection and reports a timeout error.

Two different types of queries can be sent to the recommendation module from the SMARTFASI middleware:

1. a query for a single target customer;
2. a query to manage a collective recommendation for a group of user-selected homogeneous target customers.

For each query, in addition to the customer's code, the user must provide the values for all the parameters necessary for the execution of the query. For this reason, the format of the message of type *Request* is a TCP string consisting of the following fields:

⟨**01**⟩ Internal code for command: *Request*

⟨**Querycode**⟩ Unique code associated with the query, in order to correctly associate each answer with the related request.

⟨**CustomerID**⟩ Multiple lines containing target customer ID

⟨**NULL**⟩ Null string indicating the end of the customers list

⟨**A/D**⟩ The ranking of the assets should be ascending (to consider the most frequently used assets) or descending (in case the user wants to evaluate the less frequently used assets by similar customers)

⟨**N**⟩ Number of similar customers in the ranking of *Step 1* (nullable, since it is optional)

⟨**J**⟩ Number of similar portfolios in the ranking generated by *Step 2* (nullable, since it is optional)

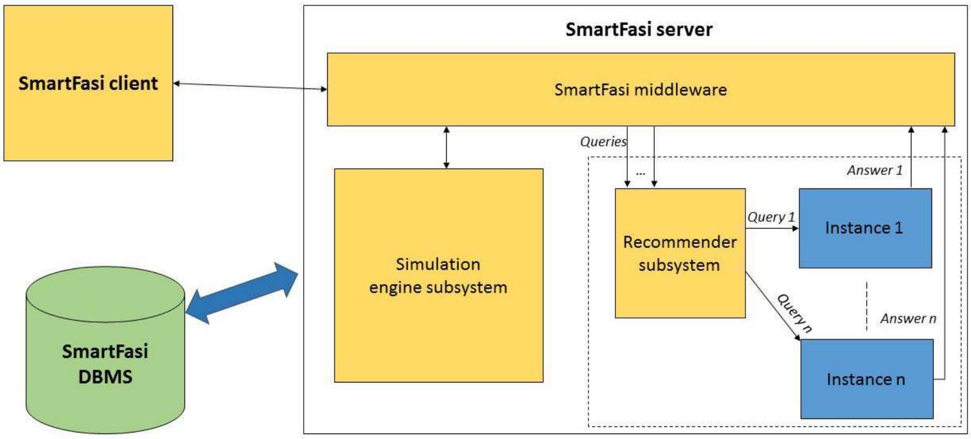


Figure 4. The SMARTFASI system's architecture

$\langle \mathbf{K} \rangle$ Number of assets to be received in response and to be shown to the user
$\langle . \rangle$ End of message

Once the server has received a query, it creates the instance aimed at computing the query result (i.e., the set of recommended financial products). The latter is then packed in a TCP *Answer* message and sent back to the SMARTFASI middleware. The result is a list of assets, each one associated with the corresponding statistics F, AP, ADC and ADP. The format of the *Answer* message is composed by the following fields:

$\langle \mathbf{A1} \rangle$ Internal code for command: *Answer*
$\langle \mathbf{Querycode} \rangle$ Unique code to correctly associate this answer to the corresponding request
$\langle \mathbf{Asset}; \mathbf{F}, \mathbf{AP}, \mathbf{ADC}, \mathbf{ADP} \rangle$ K lines containing the asset code list and their parameters
$\langle . \rangle$ End of message

In addition, the message protocol provides answer messages and codes to manage potential server malfunctions and errors (for example, to answer with an error code when a query does not contain a target customer ID).

5 Conclusion and Final Remarks

In the present paper we have described the recommendation module of a smart financial advisory system developed as part of the SMARTFASI project. Following an emerging trend [16, 14, 15], we based the recommendation strategy on Case-Based Reasoning, by defining a suitable notion of similarity among customers and their investment preferences characterized by their portfolios of financial products. The recommeded module is complementary to an asset analytical engine, based on Monte Carlo simulation.

Apart from standard recommendation of titles (potentially exploitable by both private as well as professional investors), the proposed methodology can also be exploited by financial companies during the definition of the *Asset Basket* to be proposed to the customers. The standard way of implementing the above process is to cluster customers depending on their (a-priori defined) economic/trading features, and on their adequacy to the financial products; for each cluster the so called *Investment Universe - IU* (the basket of suitable products for the cluster) is then defined and used as a basis for each proposal (see Fig. 5). This fixed strategy can be improved by resorting to similarity-based recommendation as follows: a cluster representative element is identified based on standard features (e.g., A_1, B_1, C_1) and by considering an "average" value for them[3]. The cluster representative can then play the role of the target user in the SMARTFASI recommendation engine, allowing one to extract the most (or the less) frequently used assets by customers similar to the selected profile. (Fig. 6) In this way, the recommendation engine is exploited to build alternative baskets more tailored to the actual behaviour of the customers in the considered cluster (the so called *Behavioural Investment Universes - BIU*). They may be used to update the asset baskets currently used by the company, as well as to determine the actual effectiveness of such baskets, by considering in the analysis also the appeal of some assets at the cluster level.

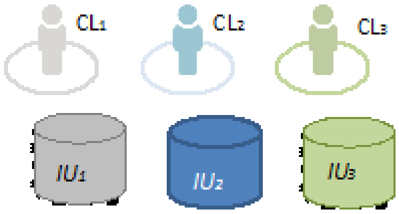


Figure 5. Fixed basket definition and proposal

More importantly, an historical analysis of such BIUs may discover specific investment trends inside each cluster, by allowing the company to implement better marketing strategies with respect to the given segment of customers. We are planning in the next future to set up an experimental plan to evaluate these kind of strategies.

ACKNOWLEDGEMENTS

The presented work has been conducted in the project SMARTFASI (funded by the ICT Innovation Cluster of the Region of Piedmont, Italy).

[3] Alternatively, we can also select several cluster representatives and to use them as a target group of customers (see Section 4.)

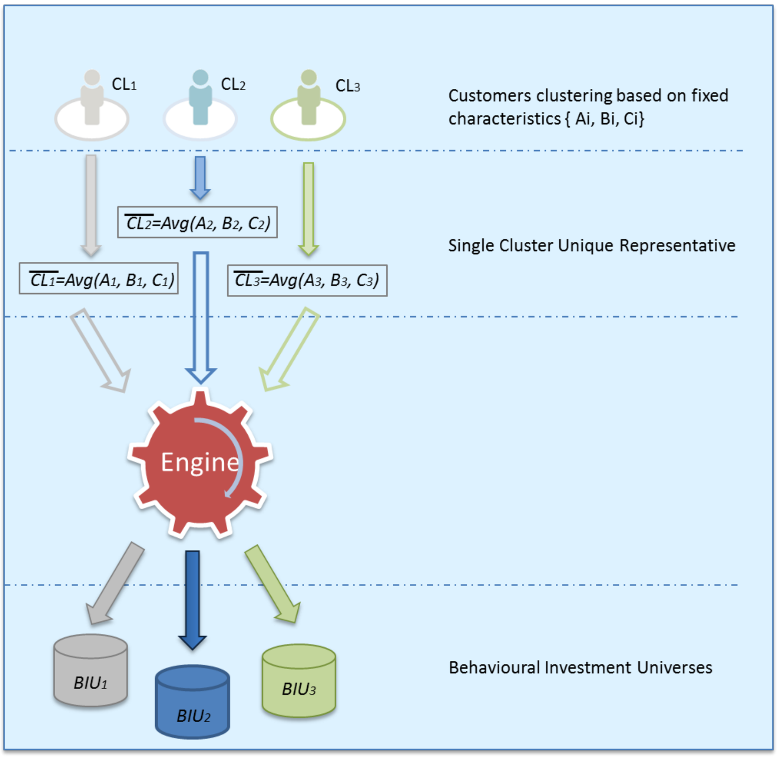

CL_1
CL_2
CL_3
Customers clustering based on fixed characteristics { Ai, Bi, Ci}
$\overline{CL_2}=Avg(A_2, B_2, C_2)$
Single Cluster Unique Representative
$\overline{CL_1}=Avg(A_1, B_1, C_1)$
$\overline{CL_3}=Avg(A_3, B_3, C_3)$
Engine
BIU_1
BIU_2
BIU_3
Behavioural Investment Universes

Figure 6. Producing Behavioural Investment Universes

REFERENCES

[1] Markets In Financial Instruments Directive 2004/39/EC. `http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32004L0039:EN:HTML`.

[2] Agnar Aamodt and Enric Plaza, 'Case-based reasoning: Foundational issues, methodological variations, and system approaches', *AI Communications*, **7**(1), 39–59, (1994).

[3] *Lazy Learning*, ed., D.W. Aha, Kluwer Academic Publishers, 1997.

[4] D. Bridge, M.H. Goeker, L. McGinty, and B. Smyth, 'Case-based recommender systems', *Knowledge Engineering Review*, **20**(3), 315–320, (2006).

[5] D. Brigo and F. Mercurio, *Interest Rate Models - Theory and Practice: With Smile, Inflation and Credit*, Springer Finance, 2006.

[6] A. Fano and S. Kurth, 'Personal choice point: Helping users visualize what it means to buy a bmw', in *Proceedings of International Conference on Intelligent User Interfaces IUI03*, (2003).

[7] A.G. Francis and A. Ram, 'Computational models of the utility problem and their application to a utility analysis of case-based reasoning', in *Proceedings of the AAAI Workshop on Knowledge Compilation and Speed-Up Learning*, (1993).

[8] J. C. Hull, *Options, Futures, and Other Derivatives (6th ed.)*, Upper Saddle River, N.J: Prentice Hall, 2006.

[9] Y. Koren and R. Bell, 'Advances in collaborative filtering', in *Recommender Systems Handbook*, eds., F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor, 145–186, Springer, (2011).

[10] Sheng-Tun Li and Hei-Fong Ho, 'Predicting financial activity with evolutionary fuzzy case-based reasoning', *Expert Systems with Applications*, **36**(1), 411 – 422, (2009).

[11] P. Lops, M. de Gemmis, and G. Semeraro, 'Content-based recommender systems: state of the art and trends', in *Recommender Systems Handbook*, eds., F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor, 73–106, Springer, (2011).

[12] F. Lorenzi and F. Ricci, 'Case-based recommender systems: a unifying view', in *Intelligent Techniques for Web Personalization*, 89–113, Springer, (2005).

[13] H. Miyamoto and T. Yonemura, 'New wave of retail asset management business from private banking to sales at bank branches', *NRI Papers*, (129). May 1,2008.

[14] C. Musto, G. Semeraro, P. Lops, M. de Gemmis, and G. Lekkas, 'A framework for personalized wealth management exploiting case-based recommender systems', *Intelligenza Artificiale*, **9**(1), 89–103, (2015).

[15] C. Musto, G. Semeraro, P. Lops, M. de Gemmis, and G. Lekkas, 'Personalized finance advisory through case-based recommender systems and diversification strategies', *Decision Support Systems*, **77**, 100–111, (2015).

[16] K. J. Oh and T. Y. Kim, 'Financial market monitoring by case-based reasoning', *Expert Systems with Applications*, **32**, 789–800, (2007).

[17] L. Portinale and P. Torasso, 'Case base maintenance in a multimodal reasoning system', *Computational Intelligence*, **17**(2), 263–279, (2001).

[18] M.M. Richer and R.O. Weber, *Case-Based Reasoning: a Textbook*, Springer, 2013.

[19] *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*, eds., G. Shakhnarovich, T. Darrell, , and P. Indyk, MIT Press, 2006.

[20] *Personalization Techniques and Recommender Systems*, eds., G. Uchyigit and M. Y. Ma, World Scientific Publ., 2008.

[21] D. Randall Wilson and Tony R. Martinez, 'Improved heterogeneous distance functions', *J. Artif. Int. Res.*, **6**(1), 1–34, (January 1997).

[22] Pavel Zezula, Giuseppe Amato, Vlastislav Dohnal, and Michal Batko, *Similarity Search: The Metric Space Approach*, volume 32 of *Advances in Database Systems*, Springer, 2006.