Expert Systems with Applications 38 (2011) 2441–2454

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A highly adaptive recommender system based on fuzzy logic for B2C e-commerce portals

Jose Jesus Castro-Schez *, Raul Miguel, David Vallejo, Lorenzo Manuel López-López

Escuela Superior de Informatica, University of Castilla-La Mancha, Paseo de la Universidad 4, Ciudad Real 13071, Spain

ARTICLE INFO

Keywords:
e-Commerce
C2C
B2C
Product selection
Recommender system
Supervised learning
Fuzzy logic
Rule-based knowledge learning

ABSTRACT

Past years have witnessed a growing interest in e-commerce as a strategy for improving business. Several paradigms have arisen from the e-commerce field in recent years which try to support different business activities, such as B2C and C2C. This paper introduces a prototype of e-commerce portal, called e-Zoco, of which main features are: (i) a catalogue service intended to arrange product categories hierarchically and describe them through sets of attributes, (ii) a product selection service able to deal with imprecise and vague search preferences which returns a set of results clustered in accordance with their potential relevance to the user, and (iii) a rule-based knowledge learning service to provide the users with knowledge about the existing relationships among the attributes that describe a given product category. The portal prototype is supported by a multi-agent infrastructure composed of a set of agents responsible for providing these and other services.

1. Introduction

The consolidation of the World Wide Web as an everyday technology has allowed the emergence of a new competitive environment where firms can develop or extend their business processes (Kowtha & Choon, 2001; Laudon & Laudon, 2005) to deal with customers from all around the world (Turban, Lee, King, & Chung, 2000). In recent years, the high competitiveness in this environment has caused great research activity focused on developing a new infrastructure aimed at supporting this new business paradigm. This is commonly known as Electronic Commerce (e-commerce), and it can be defined as any business that is electronically transacted (Cameron, 1997). e-Commerce technologies and processes have introduced new ways to do business.

Many business paradigms have arisen from the e-commerce scope. One of the most popular is C2C (Consumer-to-Consumer) e-commerce, where transactions are carried out directly by consumers who negotiate with one another to try to reach an agreement. The main feature that distinguishes C2C from other paradigms arisen from the e-commerce field is that it allows the same individual to play both the seller and the buyer roles in different transactions. One of the first C2C applications to appear, and probably the most popular, is the electronic auction (e-auction), which made possible to overcome the geographical constraints inherent to traditional auctions and allowed any individual to buy or sell goods at any time and from every corner of the world. After the success of C2C, many firms established and started to develop direct shopping activities using this infrastructure, which leaded C2C portals to be used as B2C (Business-to-Consumer) portals. In fact, according to Holahan (2008), the amount of transactions carried out through electronic auctions is slightly decreasing in recent years, whereas the number of direct shopping transactions keeps increasing steadily.

Probably because of the origin of many of the today's most popular e-commerce portals as e-auctioning sites (C2C), most of them use lexicographic descriptions and objects arrangements to manage their products catalogue. This implies that users must provide the system with key words or text strings denoting a product specific model and/or brand in order to define the search criteria. However, although this approach seems suitable to be used in an auctioning context where consumers are usually looking for very rare and exclusive goods that can be easily indexed through textual descriptions, it does not seem very appropriated to be used under the direct shopping paradigm, due to the following drawbacks: (i) lexicographic searches are prone to return irrelevant results due to the polysemy that many commonly used words have, and (ii) the specification of product features by means of text in description fields makes extremely complex the use of technologies for the automatic comparison and recommendation of products.

Besides, the direct shopping scenario clearly shows the great difference that usually exists in the uncertainty degree in the knowledge of sellers and buyers. That is, whereas sellers usually know well the features of the goods they put up for sale, buyers usually lack of a precise knowledge of what they can find, fact that often leads them to naturally specify what they want in a vague or

* Corresponding author. Tel.: +34 926295300; fax: +34 926295354.
E-mail address: JoseJesus.Castro@uclm.es (J.J. Castro-Schez).
URL: http://personal.oreto.inf-cr.uclm.es/jjcastro (J.J. Castro-Schez).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.08.033

imprecise way (Klaue, Kurbel, & Loutchko, 2001). Moreover, that lack of knowledge drives consumers to buy the most popular product, although it is possibly not the best nor the most suitable from a quality and cost perspective.

These issues lead us to propose in this work an e-commerce portal focused on the development of direct shopping activities under the B2C paradigm. The main contributions of our work are the following: (i) a hierarchical cataloguing system to arrange products according to a set of features, (ii) a product selection system (search engine) that allows the definition of vague and uncertain search criteria and returns a list of products arranged according to their relevance, and (iii) a recommender system aimed at providing consumers with knowledge about the market that allows them to define more realistic searches and to be aware of what they can buy and at what cost. For these purposes, we propose to use fuzzy logic as tool to deal with the uncertainty and vagueness in the search criteria and to communicate with the users by employing a terminology that is easily understandable for them. Besides, we have taken special care in the resulting portal usability, as this feature usually becomes seriously affected due to the large amounts of data required to specify vague or imprecise search criteria (Castro-Schez, Vallejo-Fernández, Rodriguez-Benitez, & Moreno-García, 2008).

The portal devised in this work is composed of the following components, many of them are commonly found in any current e-commerce portal (C |B)2C: a products cataloguing system, a users management system, a messaging system, a products evaluation system, an issues management system, a lexicographic search engine, a sales management system, an auctions management system, a private data management system, and a report generation system.

The remainder of this article is organised as follows. Section 2 reviews some related work relevant to develop our proposal. Section 2.1 describes the architecture of the multi-agent infrastructure that supports the proposed portal. In Section 3, some of the most important portal components are described in detail. These are the cataloguing system in Section 3.1, the product selection system in Section 3.2, and the knowledge learning system in Section 3.3. Section 4 illustrates our proposals by means of a case study. Finally, Section 5 offers a careful discussion and some concluding remarks.

2. Related work

Many software applications exist nowadays specifically developed to support the necessities of e-commerce, either from the open source community (osCommerce (open source Commerce) e-commerce solutions, 2009; PrestaShip Free Open-Source e-Commerce Software for the Web 2.0, 2009) and from private companies (CubeCart Free& Commercial Online Shopping Cart Solutions, 2009; X-Cart Shopping Cart Software & Ecommerce Solutions, 2009; Zen Cart e-commerce shopping cart software, 2009). Besides, some initiatives try to adapt some general purpose content management systems (CMS) software to fit them to the e-commerce necessities (Joomla! Open Source Content Management System, 2009).

Amazon, eBay, Ecommerce Times, Buy, Krillion, or Smallbusiness are some examples of the myriad of e-commerce portals that have reached great success in recent years. Most of them allow the development of C2C, B2C and B2B activities. E-commerce portals can be also broadly categorised as general purpose, in which any kind of product can be sold or bought, and specialised portals, which are focused on a particular category or categories of products. Besides, a new trend has emerged recently in which even small companies or individuals can open their own e-commerce portal on the web.

Nowadays, some features that most existing e-commerce portals have in common are as follows:

- *Catalogue browsing:* most portals allow the users to browse products in their catalogue. However, in many cases, the products are described and arranged according to a set of textual descriptions and just in a few cases a hierarchical catalogue is used for products arrangement.
- *Lexicographic search:* as consequence of the previous feature, most portals require the search criteria to be specified in form of text strings, denoting key words or product specific brands or models. This often causes irrelevant results to be returned due to the polysemy that many commonly used words have.
- *Advanced search:* in most cases, advanced search is focused on a set of filters applied to some products features, such as the price.
- *Mixed search:* in most cases it is allowed to specify a category or categories from the catalogue to restrict a lexicographic search.
- *Results arrangement:* in most cases, it is possible for the user to sort the returned results according to some criteria, such as the price, the users valuation, etc.

A product selection system is a service that takes as input a set of search criteria and returns a list of results arranged according to their relevance and potential interest to the user. Product selection systems are distinguished from traditional search engines in that they provide more advanced capabilities, such as the interpretation of vague or imprecise search criteria or the results clustering and classification according to their relevance. Product selection systems can be broadly classified according to the kind of products for which recommendation is offered as: (i) product selection systems for low involvement products (LIP), such as books, music albums, or films, and (ii) product selection systems for high involvement products (HIP), such as appliances, video or photo cameras, musical instruments or vehicles (Srikumar & Bhasker, 2004). For a more detailed description of what low and high involvement products mean in the marketing filed, the reader is referred to Vakratsas and Ambler (1999).

In the case of LIP, the click-to-buy rates are usually higher compared to those of HIP. Consequently, recommendations for LIP are usually offered with the help of the customer's past purchases or past searches, demographic details, or explicitly specified interests. Collaborative Filtering (CF) is one of the most widely used techniques to offer recommendations for LIP products. CF techniques try to match the customer's tastes and preferences with that of all other customers to identify those like-minded and then offer the products bought by them as recommendations. This fact allows the system to provide the users with recommendations of the form "*customers who bought an electric guitar also bought an amplifier*". Lee, Liu, and Lu (2002) and Srikumar and Bhasker (2004) offer a detailed description of a variety of product selection systems for e-commerce that use these concepts.

In the case of HIP, product selection systems are usually developed to take as input a set of product features or attributes to match against the set of products available in the database. As result, the system generates a ranking of products most likely to be of interest to the customer. Such description is provided by the user as a vector of attribute-value pairs, which is analysed by the product selection system to return a set of products sorted according to their similarity to the customer's preferences.

Product selection in the case of high involvement products results more complex compared to low involvement products. The main reasons to state this are as follows: (i) HIP require a more detailed description, as they usually have many more features than LIP have, (ii) since HIP are usually more expensive than LIP, the recommendation and decision support capabilities are more important and demanded by the customers, and (iii) since HIP receive less click-to-buy requests than LIP, bringing attention on a specific model or brand is a very interesting feature for sellers. Because of

these reasons,the remainder of this work is focused on the product selection and recommendation for HIP.

The design of a recommender system depends to a great extent on the way the products are arranged and described, that is, the cataloguing system. In Ryu (1999), a methodology for the construction of dynamic taxonomy hierarchies based on customer specified attributes is introduced. The system searches for products that satisfy the customer's preferences on the Internet. If matching products are found, they are presented to the customer. Otherwise, the similar products are presented to the customer as alternatives. A Case-Based Reasoning (CBR) approach to the product selection problem is presented in Saward and O'Dell (2000). According to it, every product in the database is represented as a case consisting of a set of attributes. The customer's preferences are also captured and represented as a case. Similarity of the customer's preferences to the product cases in the database is assessed by using a nearest neighbours approach. Products with higher similarity scores are then offered as recommendations to the customer.

One of the main issues to deal with when processing the user's search preferences is the uncertainty and vagueness with which they usually express their requirements. Besides, this is specially important in the case of high involvement products because of the large number of features they often have. Fuzzy logic (Zadeh, 1965; Zadeh, 1975) is a well validated tool to handle vagueness and uncertainty in information. Following, some proposals that make use of fuzzy logic for this purpose are described. In Yager (2003), a fuzzy logic based methodology for developing product selection systems is proposed. In this, a considerable use of fuzzy set methods is made for the representation and subsequent construction of justifications and recommendation rules. In the proposed methodology, the recommendations are solely based on the preferences of the single individual for whom recommendations are provided. Some authors call this sort of methods *reclusive methods*, as opposed to collaborative methods, such as Collaborative Filtering. In Mohanty and Bhasker (2005) the authors introduce a methodology for efficient product selection which is built upon the approach proposed in Ryu (1999).

Most of these works propose product selection systems to support consumers at the search stage. However, a very interesting feature is that the system is also able to provide guidance to the users to help them to learn what they can buy and at what cost. In Albusac, López-López, Murillo, and Castro-Schez (2008), a fuzzy logic based machine learning algorithm is proposed to infer knowledge from the market data stored in the own portal's database. The inferred knowledge is provided to users in form of fuzzy association rules that show the most relevant relationships between the set of variables used to specify the search criteria and a target variable selected by the user.

However, product selection systems based on fuzzy preferences have important usability issues compared to the traditional search engines based on lexicographic search criteria (Castro-Schez et al., 2008). These issues are a consequence of the large number of variables and data that the customers are required to provide to the system when they define a new search. In this regard, we propose in this work a product selection system able to deal with uncertain and vague search criteria and designed to impact as little as possible in the resulting portal usability and a rule-based knowledge learning system to help users in discovering which products may best match their expectancies and at what cost. These services are based on fuzzy logic concepts and techniques as well as supervised learning methods.

Finally, from a general point of view, electronic commerce involves an open and complex problem in which several entities cooperate in order to negotiate with each other. For this reason, several works have been published in the last years which propose multi-agent systems (MAS) to support negotiation activities that are carried out in e-commerce portals (Ashri, Rahwan, & Luck, 2003; Geipel & Weiss, 2007; Hindriks, Jonker, & Tykhonov, 2008; Jonker, Robu, & Treur, 2007). By using this approach, the modelling of the different parts that compose an e-commerce portal is naturally performed through intelligent agents which have the ability to communicate with other agents and to act autonomously in base of defined goals.

2.1. Portal architecture

The multi-agent architecture that gives support to the e-commerce portal proposed in this work is based on the set of standards defined by the FIPA committee (Foundation for Intelligent Physical Agents) for the development of multi-agent systems (Fig. 1). The adoption of this approach for the design of the e-commerce portal has two main goals: *(i)* to guarantee the interoperability with other multi-agent systems developed according to the FIPA guidelines, and *(ii)* to provide a set of management services common to any application developed from this general-purpose multi-agent system, such as the e-commerce application. From an abstract point of view, FIPA defines three management services that can be used by the agent platform:

- *Agent Management System* (Foundation for Intelligent Physical Agents): this component is responsible for the management of the agent platform. To do such task, it controls the life cycle of the agents which are registered with the system and it is able to instantiate new agents, modify their descriptions, or even delete them. Thus, this component also provides a white-pages service that allows the rest of elements of the multi-agent system to search for other agents by using their unique identifier.
- *Message Transport Service* (Foundation for Intelligent Physical Agents): this component provides the communication mechanism between agents of the same and different platforms. However, FIPA also enables a direct communication between agents without the need of using this intermediate communication service.
- *Directory Facilitator* (Foundation for Intelligent Physical Agents): this component, which is not mandatory, provides a yellow-pages service to the agent platform. In other words, it allows the agents registered with it to search for other agents by using a functional description. One of the more relevant characteristics of this service is the possibility for the agents to create subscriptions so that the Directory Facilitator notifies them when an agent performs a subscription, modification, or deletion.

For the users of the e-commerce portal, the fact of adopting a multi-agent approach is transparent, since they perceive the system as a set of services that can be used to provide them with a particular benefit. Next, the functionality of the main agents that compose the multi-agent architecture that gives support to the e-commerce portal is summarised:

- *Catalogue Agent*: this agent is responsible for maintaining the catalogue of the e-commerce portal and reporting about its content. The main functions of this agent are to guarantee the integrity of the catalogue and to interact with the portal administrator in order to facilitate the population of the catalogue.
- *Product Selection and Recommender Agent*: this agent supports the product selection and the knowledge learning services. It is responsible for taking a description of the user's requirements as input, for analysing it, and for searching for those products in the portal's database that best match them. The input data for this agent is provided by the user or the agent that acts as the

ambassador of such user (typically seller or buyer agents). Thus, the main functions of this agent are to manage the search process and to give recommendations to the users.

- *Auctioneer Agent*: this agent is responsible for controlling the auctions that take place in the portal. The functions of this agent consist in receiving auction requests by the human users or agents and initiating the auction process.
- *Seller Agent*: this agent can represent the human seller within the e-commerce portal and its main responsibility is to interact with the rest of agents in order to benefit its owner.
- *Buyer Agent*: this agent can represent a human buyer within the portal in order to buy products in which the user may be interested in. For that purpose, it uses the specifications previously given by the user.

3. Catalogue, product selection, and knowledge learning services

In this section, we devise our proposals for the products management and cataloguing service, the products selection and recommendation service, and the knowledge learning service in detail. These services are provided by the *Catalogue* and the *Product Selection and Recommender* agent, respectively.

3.1. Hierarchical products cataloguing service

A products catalogue $C = \{object, c_1, c_2, \ldots, c_n\}$ is composed of a set of product classes or categories c_i hierarchically organised, in which the class *object* is the root class. All categories in the catalogue are associated with each other through an *is a* transitive relationship, which ensures that $\forall c_i$, c_i *is an object*. This catalogue allows the specification of general and specific classes. Specific classes inherit the attributes and features of the general classes from which they inherit and add some particular attributes that are exclusive of the objects which belong to them.

Every product for sale in the portal must belong to at least one category $c_i \in C$. Thus, each category is composed of a set of products that belong to it $c_i = \{e^1_{c_i}, \ldots, e^n_{c_i}\}$, where each product $e^j_{c_i} \in c_i$ is conveniently described through a set of variables or attributes $V_{c_i} = \{v^1_{c_i}, \ldots, v^m_{c_i}\}$, which specify the features of the product itself. Besides, each variable $v^j_{c_i} \in V_{c_i}$ is defined through a *range of definition*, denoted as $RDV^j_{c_i} = \{x_1, x_2, \ldots, x_n\}$ or $RDV^j_{c_i} = \left[max\left(v^j_{c_i}\right), min\left(v^j_{c_i}\right)\right]$, which specifies the possible values x_k that the variable $v^j_{c_i}$ can take. The range of definition of a particular category of products c_i, denoted as RDV_{c_i}, is defined as the union of the ranges of definition of the variables that describe the products in such category V_{c_i}, i.e. $RDV_{c_i} = \{RDV^1_{c_i}, \ldots, RDV^m_{c_i}\}$. The set of variables that describe the *object* category, V_{object}, contains some general variables that are commonly used nowadays by most of the existing e-commerce portals, e.g. price, product condition (new or used), ...

The cataloguing service must be generic and flexible enough to allow either the sellers to freely define new products to put up for sale and to provide buyers with a convenient and easy-to-use mechanism to specify the features of the product they want to find. For this purpose, we propose to use the following data types for the variables that describe the products in the catalogue (Castro-Schez, Jennings, Luo, & Shadbolt, 2004).

- Continuous or numerical, such as integer or real numbers in continuous domains.
- Discrete ordered or graduated. They can take one or several values from a discrete domain and they are arranged according to some criterion.
- Discrete unordered or nominal. They can take one or several values from a discrete domain, but they are not arranged in anyway.
- Boolean or logic. They can only take two values: *true* or *false*.

Generality is one of the key issues when designing a cataloguing system for e-commerce portals. The catalogue proposed in this work is general in the following terms: (i) the system administrators can define a new product category at any time and relate it to the existing categories in the hierarchy, and (ii) it is possible to dynamically define new data types from the basic data types described above. The cataloguing system is supported by a database of which schema is shown in Fig. 2. According to it, a product category is related to one or several attributes, which are the variables that describe the products which belong to it. Note that an attribute can only belong to one product category. This fact ensures

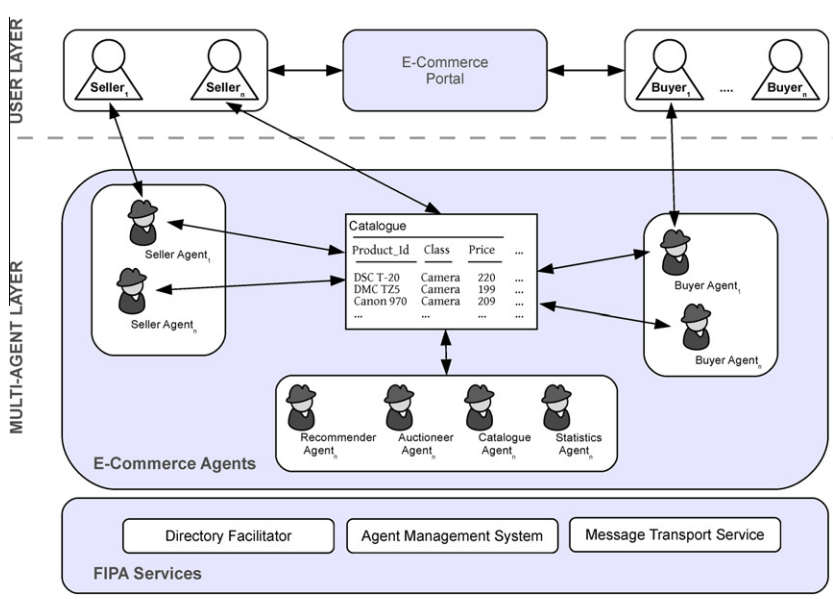

Fig. 1. General architecture of the system that supports the e-commerce portal.

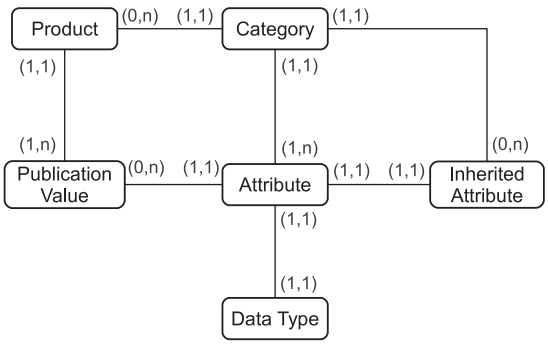

Fig. 2. Database schema that supports the proposed cataloguing system.

that there are no repeated attributes through the class hierarchy. Inheritance is the mechanism by which an attribute can be shared among different categories, although such categories must be related to each other through an *is a* relationship. Besides, in the database definition, every attribute must belong to one of the data types defined in the catalogue. This way, the system can offer instructions to the users informing them how to value each attribute. Table *data type* in the database schema shown in Fig. 2 stores the definition of all data types defined in the portal. Concerning the products managed in the catalogue, Fig. 2 shows an association between the *Category* and *Product* entities. This means that any category may contain a number of products (from zero to many). Besides, the association between the *Product* and *Publication Value* shows the instantiation of an attribute for a particular product instance.

3.2. Product selection and recommendation service

As has been aforementioned, consumers in e-commerce portals tend to specify vague and imprecise search criteria in situations where they are not aware of any specific model or brand and want to discover the existing alternatives in the market. Therefore, a product selection service able to deal with such uncertainty in the consumer's requirements is needed. Besides, as stated in Lopez-Lopez, Miguel, Albusac, and Castro-Schez (2009), the product selection systems based on fuzzy logic concepts and techniques have important usability issues due to the large amount of data that users are required to input. In this work, we have taken care of this issue and tried to simplify the vague and uncertain search criteria definition. The products selection system described in this section is composed of the following phases: (i) specification of user's requirements (search criteria), (ii) search for those products in the catalogue that best match the search criteria, and (iii) results arrangement and presentation. Next, each of these phases is described in detail.

3.2.1. Search criteria specification

The search process starts out with the definition of the search criteria. Such definition consists in selecting a category c_i in the catalogue C and value the variables that describe it according to the user's requirements and expectancies. This specification is the so-called *ideal object description*, denoted as o_i. Thus, an ideal object description o_i is defined as a subset of variables $V'_{c_i} \subseteq V_{c_i}$ from the set of all variables that describe class c_i. Besides, customers can denote how much important each variable is for them by assigning a weight $P_{v^j_{c_i}} \in [0,1]$ to them, similarly to how it is done in Castro-Schez et al. (2008).

To improve the usability of this process, a guidance service is proposed to suggest to the consumers those variables which are most commonly used by all customers that search in that category. Obviously, this is merely a decision support mechanism. It is up to the end-user to select some, all, or none of the suggested variables.

The next step is the valuation of the variables selected to define the ideal object description. Any variable $v^j_{c_i} \in V'_{c_i}$ can belong to any of the data types defined in the catalogue. Thus, it must be valued according to its particular data type and the range of definition $RDV^j_{c_i}$ associated with it. To allow for an imprecise search, the consumer can specify a *precision degree* $p \in [0,1]$, where $p = 1$ means a precise, or *crisp*, search and $p < 1$ means a fuzzy, uncertain search considering a greater degree of imprecision as the value of p becomes lower.

Due to the heterogeneity of data types in the catalogue, the system employs a common representation formalism to handle and process all variables uniformly, independently of their corresponding data types. The common representation formalism proposed in this work consists in fuzzy sets defined as trapezoidal functions with parameters (a,b,c,d), where the range $[b,c]$ represents the range of certainty and $[a,b)$ and $(c,d]$ represent the ranges of uncertainty.

Next, we describe how these trapezoids are created by the system from the consumer valuation on the variables $v^j_{c_i} \in V'_{c_i}$ selected to define the search criteria, depending on its specific data type.

- *Continuous or numerical*: These variables can be valued by selecting a single value v or a range of values $[u,v]$ from a continuous domain, along with a precision degree $p \in [0,1]$. From this data, the corresponding trapezoid is obtained as follows:
  A precision degree $p = 1$ implies a crisp search. Therefore, if a single value v has been specified, the resulting trapezoid is defined on the single numerical value v, i.e. parameters $a = b = c = d = v$. On the other hand, a range of values $[u,v]$ implies a crisp range, hence the resulting trapezoid is defined by parameters $a = b = u$ and $c = d = v$.
  Otherwise, a precision degree $p \in [0,1)$ implies a fuzzy or imprecise search. In this case, if the user specifies a single numerical value v, the resulting trapezoid is defined by parameters $b = c = v$, whereas parameters a and d are calculated as shown in Eq. (1).

$$\begin{cases} a = \dfrac{\left(\left(max\left(RDV^j_{c_i}\right) - min\left(RDV^j_{c_i}\right)\right)\cdot\beta\right)\cdot(1-m)}{m} - b \\ d = \dfrac{\left(\left(max\left(RDV^j_{c_i}\right) - min\left(RDV^j_{c_i}\right)\right)\cdot\beta\right)\cdot(1-m)}{m} + c, \end{cases} \tag{1}$$

  where $max(RDV^j_{c_i})$ and $min(RDV^j_{c_i})$ return the maximum and minimum numerical value that $v^j_{c_i} \in c_i$ can take, respectively. Eq. (1) ensures that the maximum degree of uncertainty considered by the resulting trapezoid is proportional to the amplitude of the range of definition of the variable being valued. Parameter $\beta \in [0,1]$ acts as an uncertainty delimiter, of which goal is to control how much of the range of definition amplitude is to be considered to establish parameters a and d. In our experiments, values of $\beta \simeq 0.05$ have proved to work well in most cases. Finally, the parameter m is calculated from the p value and serves to calculate the slopes of the trapezoid, as it is shown in Eq. (2).

$$m = 1 + \alpha - (1 - p), \tag{2}$$

  where α is a threshold to control those cases in which $p = 0$. In these, α serves to ensure that m never takes zero value. In our experiments, values $\alpha \simeq 0.01$ have proved to work well, as it has not alter the behaviour of the search when $p \neq 0$.
  Finally, in case that a range of numerical values $[u,v]$ is specified along with a precision degree $p \in [0,1)$, the resulting trapezoid is defined by parameters $b = u$ and $c = v$, being a and d calculated as shown in Eq. (3).

$$\begin{cases} a = \frac{(c-b)\cdot(1-m)}{m} - b, \\ d = \frac{(c-b)\cdot(1-m)}{m} + c. \end{cases} \quad (3)$$

In this case, the amount of uncertainty considered in the trapezoid is proportional to the amplitude of the trapezoid core, i.e. $(c-d)$. Thus, from the same precision degree, p, a trapezoid with a wider core considers more uncertainty than another one with a narrower core. As in the previous case, both trapezoid slopes are calculated from the m parameter as shown in Eq. (2).

- *Discrete ordered or graduated*: These variables must be valued specifying a sequence of values taken from a discrete domain. The position of the value in the sequence denotes its importance to the consumer, being the former variables preferable over the latter ones. From this specification, the system builds a trapezoid for each value in the sequence which is defined by parameters $a = b = c = d = s$, where s is the position of the value in the sequence. That is, the resulting trapezoid for the first value in the sequence is defined as $a = b = c = d = 0$, $a = b = c = d = 1$ for the second one, and so forth. A particular case of this data type are boolean variables, of which *false* value is represented by trapezoid $a = b = c = d = 0$ and *true* by $a = b = c = d = 1$ or vice versa.
- *Discrete unordered or nominal*: These variables are valued by specifying one or more values taken from a discrete domain without any order. This situation means a crisp search, so a precision degree is not required at all nor any trapezoid is needed to be built. The search process then proceeds by selecting all those products in the database that are valued with any of the selected values on the variable being considered.

3.2.2. Searching for products in the catalogue

Next, the search process proceeds to calculate the similarity of every product $e^j_{c_i} \in c_i$ in the database to the ideal object description o_i specified by the user. However, as the o_i specification is described by means of trapezoids and the values that every item $e^j_{c_i} \in c_i$ takes on each of the selected variables $V'_{c_i} \in V_{c_i}$ to define o_i are numerical values, there are two possibilities to measure the similarity of each product $e^j_{c_i}$ to o_i.

1. Convert the ideal object description values (i.e. the trapezoids that were calculated in the previous stage) into single numerical or discrete ordered values to measure the similarity between o_i and every $e^j_{c_i} \in c_i$ by using a distance measure suitable to work with lineal variables (e.g. Euclidean, Manhattan, Minkowski, etc.).

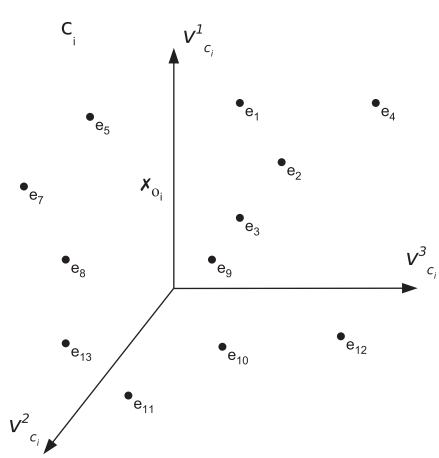

Fig. 3. Reference space on which all products that belong to category c_i are represented as points.

2. Convert the values that every product $e^j_{c_i} \in c_i$ takes on each variable $V'_{c_i} \in V_{c_i}$ selected to define o_i into trapezoids, in order to be able to compute a similarity measure between trapezoids. Let v be a single numerical or discrete ordered value, a trapezoid is defined from v as $a = b = c = d = v$.

We propose to follow the latter alternative, as it allows us to handle uniformly all variables, independently of their respective data type.

From this approach, the set of ranges of definition RDV'_{c_i} of the variables $V'_{c_i} \subseteq V_{c_i}$ selected to define o_i establishes the space of reference in which all products $e^j_{c_i} \in c_i$ and the ideal object description o_i are represented (see Fig. 3). Therefore, the similarity between any two points in the reference space is calculated by measuring the distance between them. Since the reference space has $|V'_{c_i}|$ dimensions, the distance measurement process must proceed in two steps: (i) the *normalised partial distance* between every product $e^j_{c_i} \in c_i$ and the ideal object description o_i, according to every variable $v'^j_{c_i} \in V'_{c_i}$ selected to define o_i, denoted as $d_N\left(o_i, e^k_{c_i}, v^j_{c_i}\right)$, is calculated; and (ii) the *global distance measure* between every $e^j_{c_i} \in c_i$ and o_i, denoted as $D\left(o_i, e^k_{c_i}\right)$, is calculated as the aggregation of all partial distance measures obtained in the previous step.

Since both the ideal object description o_i and each value that every product $e^j_{c_i} \in c_i$ takes on the variables in V'_{c_i} are represented as trapezoids, the *partial distance* between any $e^j_{c_i} \in c_i$ and o_i with respect to a variable $v'^j_{c_i} \in V'_{c_i}$ is calculated as the measurement of the distance between the trapezoids that represent the values that o_i and $e^j_{c_i}$ take on variable $v'^j_{c_i}$ (see Fig. 4). Nevertheless, it could be calculated employing a different measurement suitable to measure the similarity between trapezoids.

The normalisation step is performed by dividing the obtained partial distance value by the maximum distance value according to the range of definition of the variable being considered, $max(RDV^j_{c_i}) - min(RDV^j_{c_i})$. That is,

$$d_N\left(o_i, e^k_{c_i}, v^j_{c_i}\right) = \frac{d\left(o_i, e^k_{c_i}, v^j_{c_i}\right)}{max\left(RDV^j_{c_i}\right) - min\left(RDV^j_{c_i}\right)}. \quad (4)$$

In case of discrete unordered variables, the similarity value is calculated according to a heuristic that determines that the partial distance $d_N\left(o_i, e^k_{c_i}, v^j_{c_i}\right)$ is *0* if both $e^k_{c_i}$ and o_i take the same values on $v^j_{c_i}$ and *1* otherwise.

After calculating the partial distance values for every variable $v'^j_{c_i} \in V'_{c_i}$, the global distance measure between o_i and $e^j_{c_i}$ $(D(e_j, o_i))$ is calculated as the sum of all partial distance values obtained in the previous step, each one weighted by the weights given by the user in the first phase (see Eq. (5)).

$$D(e_j, o_i) = \sum_{\forall v'^j_{c_i} \in V'_{c_i}}' d_N\left(e_j, o_i, v'^j_{c_i}\right) \times P_N\left(v'^j_{c_i}\right), \quad (5)$$

where,

$$P_N\left(v'^j_{c_i}\right) = \frac{P\left(v'^j_{c_i}\right)}{\sum_{x=1}^{|V'_{c_i}|} P\left(v'^x_{c_i}\right)} \quad (6)$$

is the normalised weight associated with variable $v'^j_{c_i}$. This verifies that in every case $D(e_j, o_i) \in [0, 1]$.

3.2.3. Results arrangement and recommendation

Finally, the search process returns a list of products $e^j_{c_i} \in c_i$ arranged incrementally according to their global distance to the ideal object description o_i. Obviously, items with lower distance values

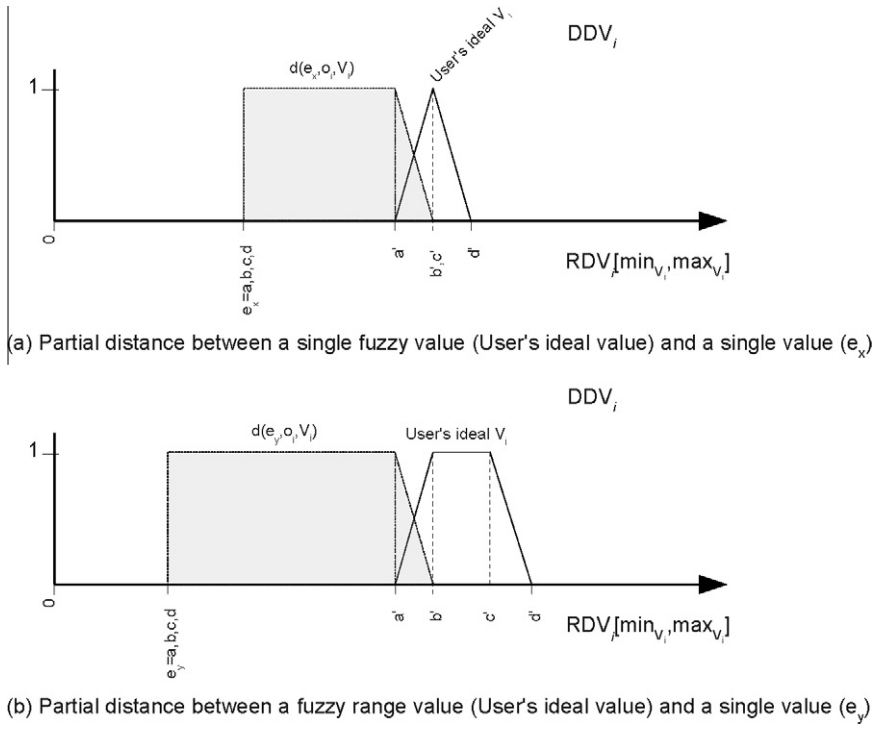

Fig. 4. The similarity between two trapezoids is calculated as the area of the trapezoid that exists between them.

are more interesting to the user than those with higher distance values, as they are closer to the user's o_i specification.

However, we need to provide users with a justification for why the system is suggesting them one item instead of another one and a raw distance value does not seem to be very meaningful for them. Therefore, we have developed an items arrangement service that classifies an item into a class depending on its global distance value to o_i. Such classes can be empty and they are not mutually exclusive. This is achieved through the definition of a new linguistic variable, *Fitness*, that can take three linguistic labels as values: *Very recommendable*, *Recommendable* and *Not much interesting*. Thus, every product $e^j_{c_i} \in c_i$ is associated with one or two of these linguistic labels by means of a process in which the global distance value $D(e_j, o_i)$ is converted into the fuzzy domain of the *Fitness* variable. The domain of definition of the *Fitness* variable is defined in the interval $[0,1]$, as in the case of the global distance, and the linguistic labels that compose it are arranged so that the lower values of the interval $[0,1]$ are assigned to those labels that denote a greater degree of interest to the user. The definition of the *Fitness* variable domain shown in Fig. 5 has proved to work well in our experiments.

3.3. Market knowledge learning

At this point, the system recommends to the customer those items in the database which are most similar to the ideal object specification o_i. However, customers are usually too much confident of the success of a search in e-commerce B2C portals. This is well known as the problem of *product overestimation*. This fact often occurs because the customer lacks of detailed knowledge about the market, i.e. what the relationships between the attributes that define the products of a category c_i are. An example of such relationship could be *the most expensive cameras are the ones with a better image quality*. Thus, the inferred rules indicate facts such as: *for the same or a slightly greater amount of money, you can buy a camera with a better image quality*. Providing the customers with such knowledge can help them to define more realistic searches according to the products that exist in the portal store, to learn the relationships among the features that describe a product category, and to find the desired product with less effort.

Such relationships could be unveiled by using model-driven techniques to analyse the products database, as it is performed in some other fields (Novak, Hoffman, & Yung, 2000). However, these techniques require a data model that must be defined a priori by the end-users. Another alternative is to use supervised learning techniques (Casillas & Martinez-Lopez, 2009; Martinez-Lopez & Casillas, 2009). These have the advantage of being data-driven, therefore they require no a priori knowledge about the particular problem in which they are applied. In our work, we propose the use of a supervised learning algorithm aimed at generating a set of association rules of the form $X \rightarrow Y$, where X and Y are sets of variables (Agrawal, Imielinski, & Swami, 1993; Casillas, Martinez-Lopez, & Martinez, 2004). These rules are inferred from the data stored in the database of the portal and describe the structural patterns found in such data. To show the quality of the inferred rules, two measurements are provided along with them: (i) number of data instances that are covered by the rule, i.e. the rule *coverage*, and (ii) the strength of the association.

In this section, we propose a supervised learning based methodology to develop a knowledge learning system aimed at unveiling the structural patterns found in the data instances stored in (C|B)2C portals databases by means of a set of association rules. Besides, such rules are required to be as comprehensible as possible. Therefore, we propose Mamdani IF-THEN rules (Mamdani & Assilian, 1975) as the representation formalism to express the discovered structural patterns in an easily understandable way. Moreover, the set of rules provided to the users must be inferred by taking into account the preferences and requirements expressed by them.

Next in this section, we describe how the training data set is, how the inference process is carried out, and how the inferred knowledge can be interpreted by the end-users.

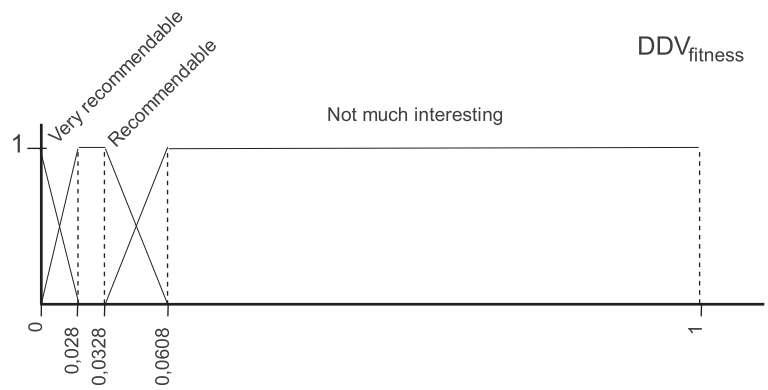

Fig. 5. Definition domain of the variable *Fitness*.

3.3.1. Training dataset

Supervised learning techniques unveil a set of structural patterns in the variables that describe a set of data instances. In our problem, a target variable is needed to be chosen from the set of all variables selected to define the ideal object description o_i. The inferred rules show the existing relationships between the rest of those variables and the selected target variable. Therefore, we need a set of data instances that belong to the category of interest in the catalogue and from which our proposed method is able to infer such relationships.

As previously described, our portal calculates the similarity of every product $e^k_{c_i} \in c_i$ in the database to the ideal object specification o_i defined by the customer. This process proceeds first obtaining the partial distance values between a product $e^k_{c_i}$ and o_i according to each variable $v^j_{c_i} \in V'_{c_i}$, denoted as $d_N\left(e^k_{c_i}, o_i, v^j_{c_i}\right)$, and then obtaining the total distance value between $e^k_{c_i}$ and o_i by aggregating the partial distances obtained before. The result of the first step is a table of partial distances in which the rows are all the products $e^k_{c_i} \in c_i$ stored in the database and the columns are all the variables $v^j_{c_i} \in V'_{c_i}$ selected to define the o_i specification. At the intersection of a row k with a column j, the value $x_{k,j}$ specifies the partial distance between the product $e^k_{c_i}$ and the o_i specification according to variable $v^j_{c_i}$ $(1 \leqslant k \leqslant |c_i|$ and $1 \leqslant j \leqslant |V'_{c_i}|)$.

From this data, we aim at learning rules of the form: IF $v^1_{c_i}$ is EQUAL or MORE or MUCH MORE and $v^2_{c_i}$ is EQUAL and ... THEN $v^x_{c_i}$ is MUCH LESS. This rule can be converted into a recommendation sentence of the form: For MUCH LESS $v^x_{c_i}$ you may achieve EQUAL, MORE, or MUCH MORE $v^1_{c_i}$ and EQUAL $v^2_{c_i}$ and .... Where the values EQUAL, MORE, LESS, MUCH MORE, and MUCH LESS are linguistic labels that compose the domain of definition of a linguistic variable defined on the interval [0,1], as $(d_N(x, y, v^j_{c_i})) \in [0, 1]$.

However, the similarity measure proposed in Castro-Schez, Jimenez, Moreno, and Rodriguez (2005) used to calculate the partial distances has some problems. As illustrated in Fig. 6, the trapezoids corresponding to the value of the variable *price* on products $e^x_{c_i}$ and $e^y_{c_i}$ are both the same distance away from the trapezoid corresponding to the value of the variable *price* defined on the o_i specification. According to the measure proposed in Castro-Schez et al. (2005), both distance values are equal, fact that makes impossible to distinguish which product has a lower value and which a greater value, because the normalised partial distance takes values within the interval [0,1].

We have solved this problem by converting the partial distances to a similarity measure that takes values in the interval [−1,+1]. In this case, a value near to zero (whether it is negative or positive) means that the product value on that variable is very similar to the valuation defined in the o_i specification. On the other hand, as the similarity value between the product value and the o_i value on variable $v^j_{c_i}$ increases, it means that the product value is greater than the o_i value. Conversely, as the similarity value decreases towards −1, it means that the product value is less than the o_i value.

Converting the partial distances values $x_{k,j} \in [0,1]$ into the similarity values $x_{k,j} \in [-1,+1]$ is quite a trivial process and it is not computationally involved at all. First, the partial distance value $x_{k,j}$ between product $e^k_{c_i}$ and o_i according to variable $v^j_{c_i}$ is retrieved. Then, the actual values that the example $e^k_{c_i}$ and the o_i specification take on variable $v^j_{c_i}$ are compared to each other. If the value taken by $e^k_{c_i}$ is less than that taken by o_i, the partial distance value $x_{k,j}$ is multiplied by −1 and by +1 otherwise. Thus, values near zero, whether they are positive or negative, mean that the product has a similar valuation to that established in the ideal object description, whereas such value becomes greater or lower as the product valuation is greater or lower than that established in o_i, respectively.

Table 1 shows the form of the table of partial distances. Values of $x_{k,j}$ near to zero mean that the product $e^k_{c_i}$ is very similar to o_i according to the variable $v^j_{c_i}$, so from a partial distances table we are able to infer which attributes make a product $e^k_{c_i}$ similar or dissimilar to the customer's requirements expressed in o_i. This dataset is the raw data from which the algorithm presented in next section infers a set of fuzzy association rules.

3.3.2. Rules inference

We propose a MISO (Multiple Inputs Single Output) system of which model is described by the set of training data instances $\theta = \{e_1, e_2, \ldots, e_m\}$, where m is the number of data instances in the training dataset. Each e_i is represented in the following form:

$$e_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,n} : x_{i,t})$$

being $x_{k,j} = d_N(e^k_{c_i}, o_i, v^j_{c_i})$. Note that all variables that are considered at this stage are those which were selected by the end-user to define o_i.

The objective in this step is to approximate the following function:

$$\Omega : d_N\left(e^k_{c_i}, o_i, v^1_{c_i}\right) \times d_N\left(e^k_{c_i}, o_i, v^2_{c_i}\right) \times \cdots \times d_N\left(e^k_{c_i}, o_i, v^j_{c_i}\right) \rightarrow d_N\left(e^k_{c_i}, o_i, v^t_{c_i}\right),$$

which models the existing relationships between the variables that describe the data instances by means of a set of fuzzy association rules. These are obtained through a generalisation process based on the data instances $e_i \in \theta$. The proposed algorithm generates rules of the form:

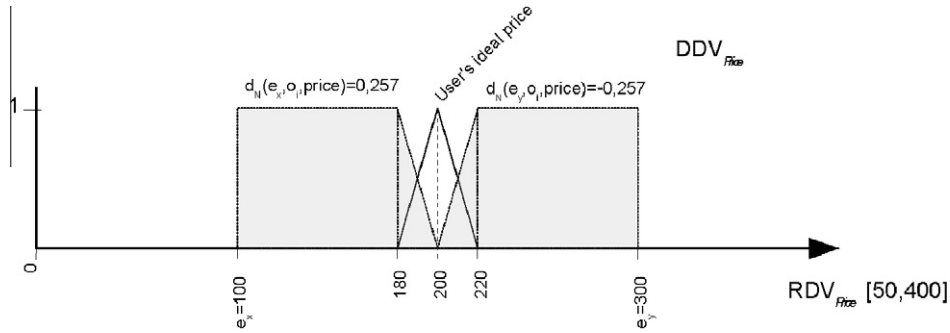

Fig. 6. *Price* variable value in the o_i specification and *Price* values taken by two different products $e^x_{c_i}$ y $e^y_{c_i}$. The partial distances $d_N(e^x_{c_i}, o_i, v^j_{c_i})$ and $d_N(e^y_{c_i}, o_i, v^j_{c_i})$ return the same absolute value, although the first is negative to express the fact that the *Price* on $e^x_{c_i}$ is lower than that established on o_i.

Table 1
Table of partial distances between each product $e^k_{c_i}$ and the o_i specification according to each variable $v^j_{c_i} \in V'_{c_i}$.

$e^k_{c_i}$	$v^1_{c_i}$	$v^2_{c_i}$	...	$v^m_{c_i}$
$e^1_{c_i}$	$x_{1,1}$	$x_{1,2}$	...	$x_{1,m}$
$e^2_{c_i}$	$x_{2,1}$	$x_{2,2}$	...	$x_{2,m}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$e^n_{c_i}$	$x_{n,1}$	$x_{n,2}$	...	$x_{n,m}$

$$\text{IF } v^1_{c_i} \text{ is } ZD_{1,dist} \text{ AND } v^2_{c_i} \text{ is } ZD_{2,dist} \text{ AND } \ldots \quad \text{THEN } v^t_{c_i} \text{ is } x,$$

which gives us information about the relationships between the variables $v^j_{c_i}$ in the antecedent part of the rule and the target variable $v^t_{c_i}$ in the consequent part of it.

The element $ZD_{j,dist}$ is a set of values taken from $\mathscr{P}(DDV_{dist})$, where DDV_{dist} is the domain of definition of variable *dist*. Fig. 7 shows the domain of definition of the variable *dist*.

$$DDV_{dist} = \{\text{MUCH LESS, LESS, EQUAL, MORE, MUCH MORE}\}.$$

Each variable $v^t_{c_i}$ used as a target attribute defines a new Ω function. Obviously, this means that the algorithm must be run every time a different target variable $v^t_{c_i}$ is selected, since this defines a new Ω function. Therefore, to provide customers with useful knowledge about the market, the variables used as target attributes must be those which are more relevant to the customer. This is automatically managed by our system by selecting those variables which have been assigned a greater weight $P_{v^j_{c_i}}$ as target attributes.

The main idea of the proposed algorithm is to generate a set of initial rules that are very particular from the set of training data instances to later submit them to a generalisation process aimed at building rules that can cover many data instances. The proposed method is an adaptation of the algorithm introduced in Castro, Castro-Schez, and Zurita (1999), which proceeds as follows:

1. Convert the training data instances into initial rules. In these, the values of each input variable $x_{k,j}$, as well as the value of the target variable $x_{k,t}$, are converted into the fuzzy domain according to the domain of definition of the *dist* variable, DDV_{dist}. The result of this step is a set of initial rules R_i

   $$A^{1,k}, \ldots, A^{n,k} : A^{t,k},$$

   where $A^{j,k} = max_k\{\mu_k(x_{k,j})\}$ is that linguistic label from the domain of definition DDV_{dist} of which membership value of $x_{k,j}$ is maximum.
   This rule can be abbreviated as:

   $$R_i: \text{ IF } v^1_{c_i} \text{ is } ZD_{1i} = \{A^{1,k}\} \text{ AND } \ldots \text{AND } v^n_{c_i} \text{ is } ZD_{ni} = \{A^{n,k}\} \quad \text{THEN } v^t_{c_i} \text{ is } A^{t,k}.$$

   Thus, this stage merely performs a translation of the values $x_{k,j} \in [-1,+1]$ in the partial distances tables to the domain of definition DDV_{dist}.
2. Build general rules from the initial ones. The generated general rules are of the following form:

   $$R_i: \text{ IF } v^1_{c_i} \text{ is } ZD_{1i} \text{ AND } v^2_{c_i} \text{ is } ZD_{2i} \text{ AND } \ldots \text{AND } v^n_{c_i} \text{ is } ZD_{ni} \quad \text{THEN } v^t_{c_i} \text{ is } A^{t,k},$$

   where each ZD_{ij} is a set of values taken from DDV_{dist} and disjunctively associated with one another. For this purpose, the algorithm tries to subsume the initial rules in general rules to obtain the set of definitive rules which are provided to the end-user. For a more detailed explanation of this process, the reader is referred to Albusac et al. (2008) and Castro et al. (1999). Briefly, the generalisation process proceeds as follows:
   (a) Take a rule R_i: $(ZD_{1i}, ZD_{2i}, \ldots, ZD_{ni}: A^{t,k})$ from the set of initial rules.

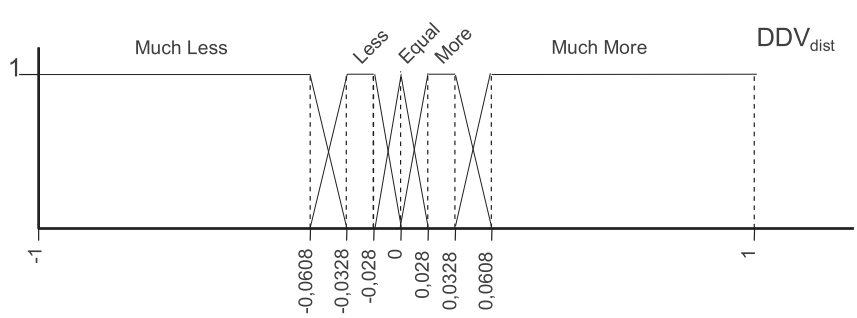

Fig. 7. Definition domain of the variable *dist*.

(b) If the taken rule subsumes in some rule from the set of definitive rules, ignore that rule and go to step (c).
A rule R_i: $(ZD_{1i}, ZD_{2i}, \ldots, ZD_{ni}: A^{t,k})$ subsumes in some other rule $R_j : (ZD_{1j}, ZD_{2j}, \ldots, ZD_{nj} : A^{t,k'})$ if: $ZD_{1i} \subseteq ZD_{1j}$, $ZD_{2i} \subseteq ZD_{2j}, \ldots, ZD_{ni} \subseteq ZD_{nj}$ and $A^{t,k} = A^{t,k'}$.

(c) For each variable $v^j_{c_i}$ in rule R_i:
  (i) For each label $A^{x,y} \in DDV_{dist}$, such that $A^{x,y} \notin ZD_{ji}$.
    - Amplify R_i with $A^{x,y}$ such that $ZD_{ji} = ZD_{ji} \cup \{A^{x,y}\}$, if possible.

(d) Add the amplified rule to the set of definitive rules.

(e) Go to step (c) while there are unconsidered rules in the set of initial rules. Otherwise, END.

Concerning the amplification process, note that a rule R_i can be amplified into another one R'_i with a new label if the following constraints are satisfied:

(a) There are no rules $R_j : (ZD_{1j}, ZD_{2j}, \ldots, ZD_{nj} : A^{t,k'})$ in the set of initial rules that verify $ZD_{1j} \subseteq ZD_{1i'}$, $ZD_{2j} \subseteq ZD_{2i'}, \ldots, ZD_{nj} \subseteq ZD_{ni'}$ and $A^{t,k} \neq A^{t,k'}$.

(b) R_i can be amplified with a new label $A^{x,y}$ on a variable $v^j_{c_i}$ if another label $A^{x,k} \in ZD_{ji}$ exists that verifies $d_N(A^{x,y}, A^{x,k}, v^j_{c_i}) \leqslant \alpha$, being α a threshold defined a priori by the administrators.

(c) There exists some evidence in the set of data instances that agrees with the amplification.

3.3.3. Rules interpretation

After the inference step, the algorithm produces a set of rules of the form:

$$R_i : \text{IF } v^1_{c_i} \text{ is } \{\text{Equal, More}\} \text{ AND } v^2_{c_i} \text{ is } \{\text{Less}\} \text{ AND } v^3_{c_i} \text{ is } \{\text{More, Much More}\} \text{ THEN } v^t_{c_i} \text{ is } \{\text{Less}\} \ < x\% > .$$

From the interpretation of the rule R_i, the following conclusions can be drawn: "*There are some products in the database with a slightly lower value of $v^t_{c_i}$ than that specified in o_i, an equal or slightly greater value of $v^1_{c_i}$ than that specified in o_i, a slightly lower value of $v^2_{c_i}$ than that specified in o_i, and a greater value of $v^3_{c_i}$ than that specified in o_i*". The term < x%> specifies the proportion of data instances that belong to class c_i that are covered by the rule.

Although the generated rules are self-explanatory per se, we have developed an interpretation service to express the inferred knowledge to the end-user by means of sentences like:

For Less $v^t_{c_i}$ you can achieve Equal or More $v^1_{c_i}$ AND Less $v^2_{c_i}$ AND More or Much More $v^3_{c_i}$.

This is easily managed by reordering the elements of the rule and changing the terminology, employing the values taken by the variables in the inferred rules, which are meaningful to the end-users.

4. Case study: searching and selecting the user's ideal product

This section describes how the e-commerce systems works when a user searches for a desired product o_i. Particularly, this description is focused on the search and selection processes provided by the e-commerce portal that is presented in this paper. The product to be acquired by the user is a digital camera. Within this context, the portal hosts a database with a category named *Digital Cameras*, which is composed of 119 items. This category is described by means of 51 variables, such as *price*, *model*, *resolution*, etc.

The input given by the user to the system consists in a vague description of the desired camera. To do that, the user selects the variables (*price*, *weight*, and *length*) and specifies the desired values, i.e. $V'_{c_i} = \{price, weight, length\}$, which is a subset of the set of variables that define the digital cameras within the portal. This description is named *ideal object description*, o_i (see Section 3.2.1).

Let us suppose that the given specification (o_i) is as follows:

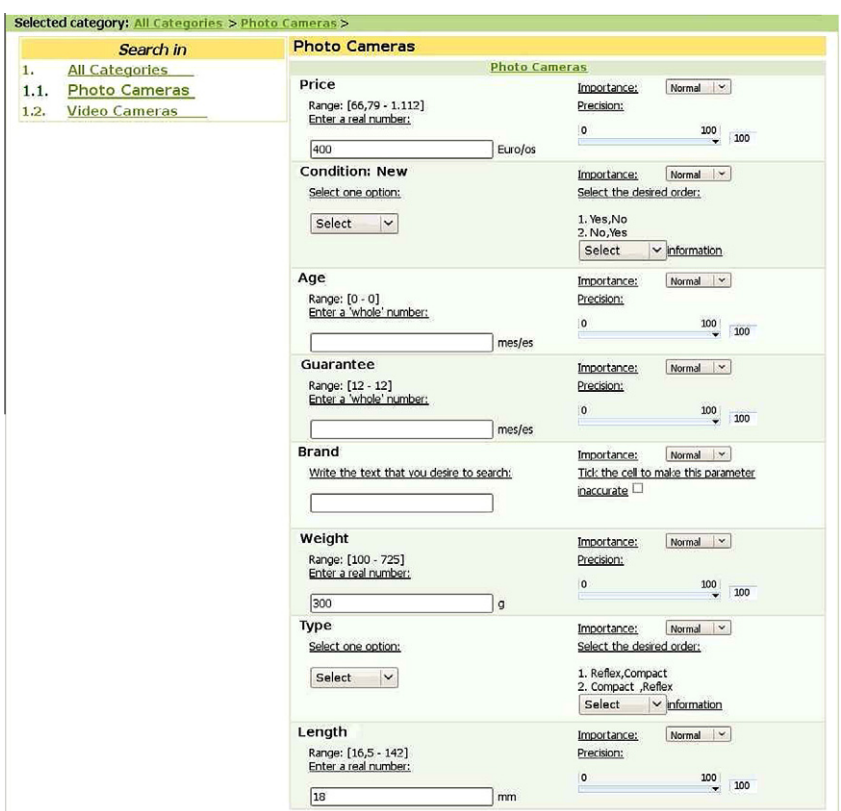

Fig. 8. Search criteria definition in the e-Zoco B2C portal prototype.

Table 2
Training set with partial distances between each product $e^k_{c_i}$ and the specification o_i given by the user regarding each variable $v^j_{c_i} \in V'_{c_i}$.

$e^k_{c_i}$	Price	Weight	Length	Global distance
Sony Cyber-shot DSC-T200	−0.185	−0.181	0.015	0.127
Panasonic Lumix DMC TZ5	−0.1	−0.136	0.15	0.129
Canon Ixus 970 IS	−0.107	−0.231	0.074	0.137
Canon PowerShot A650 IS	−0.109	0	0.303	0.137
Olympus MJU 770 SW	−0.184	−0.231	0.019	0.145
Panasonic Lumix DMC-FX100EG-S	−0.143	−0.242	0.054	0.147
Sony Ericsson Cyber-SHOT DSC-T2B	−0.153	−0.272	0.016	0.147
Casio Exilim EX-Z 1080	−0.252	−0.175	0.02	0.149
Casio Exilim EX-Z1200	−0.181	−0.236	0.034	0.15
Panasonic Lumix DMC-TZ3	−0.206	−0.108	0.15	0.155
Canon IXUS 950 IS	−0.19	−0.215	0.066	0.157
Canon Digital IXUS 860 IS	−0.181	−0.231	0.062	0.158
Nikon Coolpix P5100	−0.134	−0.159	0.182	0.158
Panasonic Lumix DMC-FX55	−0.191	−0.25	0.039	0.16
Canon Digital IXUS 75/ Powershot SD750/	−0.211	−0.27	0.011	0.164
Ricoh Caplio R7	−0.2	−0.263	0.041	0.168
Sony Cybershot DSC-W200	−0.182	−0.252	0.073	0.169
Nikon Coolpix S200	−0.238	−0.279	0.003	0.173
Olympus MJU 840	−0.203	−0.271	0.047	0.173
Olympus FE-300	−0.2	−0.295	0.031	0.175
Panasonic Lumix DMC-FX33	−0.23	−0.268	0.031	0.176
Casio Exilim EX-Z100	−0.203	−0.301	0.024	0.176
Sony Ericsson Cyber-SHOT DSC-T20	−0.219	−0.276	0.037	0.177
Sony Cybershot DSC-W90	−0.214	−0.28	0.038	0.177
Casio Exilim EX-Z200	−0.21	−0.288	0.036	0.178
Canon Digital IXUS 70	−0.248	−0.279	0.01	0.179
Sony Ericsson Cyber-SHOT DSC-W130	−0.218	−0.282	0.038	0.179
Fujifilm FinePix F480	−0.252	−0.255	0.039	0.182
Olympus FE-230/ X-790	−0.247	−0.311	−0.011	0.19
Sony Ericsson Cyber-SHOT DSC-W80S	−0.252	−0.28	0.038	0.19

- *Price* = 400,
- *Weight* = 300,
- *Length* = 18

with a 95% precision and a regular weight (0,33). Fig. 8 shows how the definition of these preferences is performed in our portal prototype.

Then, the internal representation of o_i is as follows:

- *Price* = (397.31; 400; 400; 402.69),
- *Weight* = (298.39; 300; 300; 301.61),
- *Length* = (17.67; 18; 18; 18.32).

Once the o_i description has been given, the similarity of each camera in the catalogue to it is calculated. This process is depicted in Tables 2 and 3 by taking into account the absolute value of the shown data.

The search or product selection process returns a list of digital cameras ordered according to their values of global distance. This list is then divided as discussed in Section 3.2.3, grouping the products depending on the linguistic variable *Fitness*:

- Very recommendable:
- Recommendable:
- Not much interesting: {Sony Cyber-shot DSC-T200, Panasonic Lumix DMC TZ5, Canon Ixus 970 IS, Canon PowerShot A650 IS, Olympus MJU 770 SW, Panasonic Lumix DMC-FX100EG-S, Sony Ericsson Cyber-SHOT DSC-T2B, Casio Exilim EX-Z 1080, Casio Exilim EX-Z1200, Panasonic Lumix DMC-TZ3, Canon IXUS 950 IS, Canon Digital IXUS 860 IS, Nikon Coolpix P5100, Panasonic Lumix DMC-FX55, Canon Digital IXUS 75/ Powershot SD750/, Ricoh Caplio R7, Sony Cybershot DSC-W200, Nikon Coolpix S200, Olympus MJU 840, Olympus FE-300, Panasonic Lumix DMC-FX33, Casio Exilim EX-Z100, Sony Ericsson Cyber-SHOT DSC-T20, Sony Cybershot DSC-W90, Casio Exilim EX-Z200, Canon Digital IXUS 70, Sony Ericsson Cyber-SHOT DSC-W130, Fujifilm FinePix F480, Olympus FE-230/ X-790, Sony Ericsson Cyber-SHOT DSC-W80S, Kodak EasyShare V1003, Casio Exilim EX-Z80, Olympus FE-280, Fujifilm FinePix Z10FD, Panasonic Lumix DMC-FX10, Pentax Optio S10, Panasonic Lumix DMC-FX12, Canon PowerShot A720 IS, Kodak Easyshare M753, Canon PowerShot A590 IS, Olympus FE-310/ X-840, BenQ DC C740I, Nikon Coolpix L15, Canon PowerShot A570 IS, Panasonic Lumix DMC-FZ8, Olympus FE-210, Canon Powershot A560, Kodak Easyshare C813, Kodak EasyShare C613, Fujifilm FinePix S5800, Olympus SP-550 UZ, Kodak Easyshare C713, Panasonic Lumix DMC-FZ18, Fujifilm FinePix S8000FD, Canon PowerShot S5 IS, Sony Cybershot DSC-H9, Toshiba Camileo 6IN1 PX1333E-1CAM, Sony Cyber-shot DSC-T70,Panasonic Lumix DMC-FZ50.}

As can be appreciated, the specification provided by the user does not match with any existing product in the database that

Table 3
Training set with partial distances between each product $e^k_{c_i}$ and the specification o_i given by the user regarding each variable $v^j_{c_i} \in V'_{c_i}$ (cont. from Table 2).

$e^k_{c_i}$	Price	Weight	Length	Global distance
Kodak EasyShare V1003	−0.267	−0.252	0.054	0.191
Casio Exilim EX-Z80	−0.254	−0.319	0.007	0.193
Olympus FE-280	−0.268	−0.306	0.007	0.194
Fujifilm FinePix Z10FD	−0.276	−0.303	0.005	0.195
Panasonic Lumix DMC-FX10	−0.259	−0.279	0.047	0.195
Pentax Optio S10	−0.261	−0.303	0.023	0.195
Panasonic Lumix DMC-FX12	−0.268	−0.279	0.047	0.198
Canon PowerShot A720 IS	−0.248	−0.159	0.189	0.199
Kodak Easyshare M753	−0.276	−0.295	0.039	0.203
Canon PowerShot A590 IS	−0.246	−0.199	0.18	0.208
Olympus FE-310/X-840	−0.272	−0.255	0.098	0.208
BenQ DC C740I	−0.318	−0.255	0.062	0.212
Nikon Coolpix L15	−0.27	−0.295	0.09	0.218
Canon PowerShot A570 IS	−0.277	−0.199	0.196	0.224
Panasonic Lumix DMC-FZ8	−0.185	0.015	0.485	0.228
Olympus FE-210	−0.306	−0.284	0.098	0.229
Canon Powershot A560	−0.287	−0.215	0.199	0.233
Kodak Easyshare C813	−0.298	−0.26	0.152	0.237
Kodak EasyShare C613	−0.315	−0.26	0.152	0.242
Fujifilm FinePix S5800	−0.23	0.01	0.498	0.246
Olympus SP-550 UZ	−0.162	0.103	0.477	0.247
Kodak Easyshare C713	−0.315	−0.26	0.174	0.249
Panasonic Lumix DMC-FZ18	−0.104	0.095	0.556	0.252
Fujifilm FinePix S8000FD	−0.137	0.175	0.487	0.266
Canon PowerShot S5 IS	−0.118	0.239	0.474	0.277
Sony Cybershot DSC-H9	−0.128	0.17	0.538	0.279
Toshiba Camileo 6IN1 PX1333E-1CAM	−0.286	−0.252	0.421	0.319
Sony Cyber-shot DSC-T70	−0.193	1	0.171	0.455
Panasonic Lumix DMC-FZ50	−0	0.588	0.987	0.525

Table 4
Set of initial rules for the price variable.

R_i	Price	Length	Price	Matches	Percentage
R_1	MUCH LESS	MORE	MUCH LESS	23	38.9
R_2	MUCH LESS	MUCH MORE	MUCH LESS	14	23.7
R_3	MUCH LESS	EQUAL	MUCH LESS	12	20.3
R_4	MUCH MORE	MUCH MORE	MUCH LESS	6	10.1
R_5	EQUAL	MUCH MORE	MUCH LESS	3	5
R_6	MUCH MORE	MUCH MORE	EQUAL	1	1.6

Table 5
Set of initial rules for the weight variable.

R_i	Price	Length	Price	Matches	Percentage
R_1	MUCH LESS	MORE	MUCH LESS	23	38.9
R_2	MUCH LESS	MUCH MORE	MUCH LESS	14	23.7
R_3	MUCH LESS	EQUAL	MUCH LESS	12	20.3
R_4	MUCH LESS	MUCH MORE	MUCH MORE	6	10.1
R_5	MUCH LESS	MUCH MORE	EQUAL	3	5.0
R_6	EQUAL	MUCH MORE	MUCH MORE	1	1.6

Table 6
Set of initial rules for the length variable.

R_i	Price	Weight	Length	Matches	Percentage
R_1	MUCH LESS	MUCH LESS	MORE	23	38.9
R_2	MUCH LESS	MUCH LESS	MUCH MORE	14	23.7
R_3	MUCH LESS	MUCH LESS	EQUAL	12	20.3
R_4	MUCH LESS	MUCH MORE	MUCH MORE	6	10.1
R_5	MUCH LESS	EQUAL	MUCH MORE	3	5
R_6	EQUAL	MUCH MORE	MUCH MORE	1	1.6

Table 7
Set of general rules for the price variable.

R_i	Weight	Length	Price	Matches	Percentage
R_1	MUCH LESS	EQUAL, MORE, MUCH MORE	MUCH LESS	49	83
R_2	MUCH MORE	MUCH MORE	MUCH LESS	6	10.1
R_3	EQUAL	EQUAL, MORE, MUCH MORE	MUCH LESS	3	5
R_4	MUCH MORE	MUCH MORE	EQUAL	1	1.6

meets the given restrictions, since there is no camera that belongs to the groups *very recommendable*, or *recommendable*. This is a fair example of market ignorance. Next, the learning process to generate rules to guide the user when providing more realistic product specifications by taking into account the market is discussed.

Table 8
Set of general rules for the weight variable.

R_i	Price	Length	Weight	Matches	Percentage
R_1	MUCH LESS	EQUAL, MORE	MUCH LESS	35	59.3
R_2	MUCH LESS	MUCH MORE	MUCH LESS	14	23.7
R_3	MUCH LESS	MUCH MORE	MUCH MORE	6	10.1
R_4	MUCH LESS	MUCH MORE	EQUAL	3	5
R_5	EQUAL	MUCH MORE	MUCH MORE	1	1.6

Table 9
Set of general rules for the length variable.

R_i	Price	Weight	Length	Matches	Percentage
R_1	MUCH LESS	MUCH LESS	MORE	23	38.9
R_2	MUCH LESS	MUCH LESS	MUCH MORE	14	23.7
R_3	MUCH LESS	MUCH LESS	EQUAL	12	20.3
R_4	MUCH LESS	MUCH MORE	MUCH MORE	6	10.1
R_5	MUCH LESS	EQUAL	MUCH MORE	3	5
R_6	EQUAL	MUCH MORE	MUCH MORE	1	1.6

Table 10
Set of final rules for the price variable.

R_i	Weight	Length	Price	Matches	Percentage
R_1	MUCH LESS	EQUAL, MORE, MUCH MORE	MUCH LESS	49	83

From the information obtained about the distances in the searching process, the generated set of initial rules is shown in Tables 4–6. Fig. 9 shows the set of association rules that were inferred from the search criteria and the market data stored in the portal's database, and the returned results clustered according to their similarity to the search criteria.

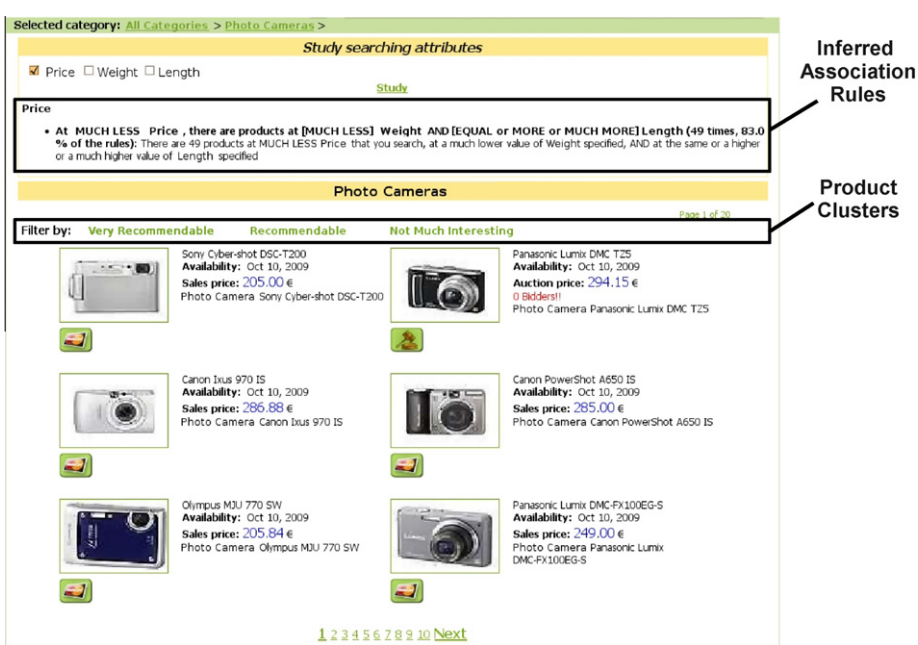

Fig. 9. Results returned by the system in response to the defined search criteria. On the top half, the inferred association rules provide guidance to the user in order to define search criteria more realistic according to the market data stored in the portal's database. On the bottom half, the set of results are clustered in different groups according to their similarity to the search criteria.

Fig. 10. Results returned by the system in response to the redefinition of search criteria according to the guidance provided by the inferred knowledge.

The generalisation process applied to the initial rules generates the set of general rules is shown in Tables 7–9. From this set, the following rules are used to give support for an advice or orientation.

These rules are converted by the portal into the following advice, see Table 10:

- *For much less price than the specified by you, you can find products with less weight and the same, little more, or much more length than the specified ones (49 products support this recommendation).*

The user modifies his/her initial specification by taking into account these advices and providing a new specification:

- *Price* = 200,
- *Weight* = 180,
- *Length* = 18

with a 95% precision and a regular weight (0,33).
Finally, the system recommends the following products:

- Very recommended: {Sony Cyber-shot DSC-T200}.
- Recommended: {Olympus MJU 770 SW, Casio Exilim EX-Z1200, Canon IXUS 950 IS, Casio Exilim EX-Z 1080 Panasonic Lumix DMC-FX55, Canon Digital IXUS 860 IS, Canon Digital IXUS 75/ Powershot SD750/, Ricoh Caplio R7, Sony Ericsson Cyber-SHOT DSC-T2B, Nikon Coolpix S200, Olympus MJU 840, Sony Cyber-shot DSC-W200}.

- Not much interesting:{Olympus FE-300, Panasonic Lumix DMC-FX33, Casio Exilim EX-Z100, Sony Ericsson Cyber-SHOT DSC-T20, Sony Cybershot DSC-W90, Panasonic Lumix DMC-FX100EG-S, Casio Exilim EX-Z200, Canon Digital IXUS 70, Sony Ericsson Cyber-SHOT DSC-W130, Fujifilm FinePix F480, Olympus FE-230/ X-790, Sony Ericsson Cyber-SHOT DSC-W80S, Kodak EasyShare V1003, Canon Ixus 970 IS, Casio Exilim EX-Z80, Olympus FE-280, Fujifilm FinePix Z10FD, Panasonic Lumix DMC-FX10, Pentax Optio S10, Panasonic Lumix DMC-FX12, Kodak Easyshare M753, Canon PowerShot A590 IS, Olympus FE-310/ X-840, Panasonic Lumix DMC-TZ3, BenQ DC C740I, Nikon Coolpix P5100, Nikon Coolpix L15, Canon PowerShot A720 IS, Canon PowerShot A570 IS, Panasonic Lumix DMC TZ5, Olympus FE-210, Canon Powershot A560, Kodak Easyshare C813, Kodak EasyShare C613, Kodak Easyshare C713, Canon PowerShot A650 IS, Toshiba Camileo 6IN1 PX1333E-1CAM, Panasonic Lumix DMC-FZ8, Fujifilm FinePix S5800, Olympus SP-550 UZ, Fujifilm FinePix S8000FD, Panasonic Lumix DMC-FZ18, Sony Cybershot DSC-H9, Canon PowerShot S5 IS, Sony Cyber-shot DSC-T70, Panasonic Lumix DMC-FZ50}.

Fig. 10 shows the results in the cluster *Very interesting* that were returned in response to the redefinition of search criteria according to the guidance provided by the inferred set of association rules.

This case study can be evaluated by using the e-Zoco portal prototype, which is available at e-Zoco (B|C)2C e-Marketplace (2010).

5. Conclusions

C2C e-commerce web portals became very popular in the last decade due to the popularity reached by some applications, e.g. e-auctions. This fact leaded many companies which developed C2C e-commerce activities to extend their business processes to allow direct shopping activities (B2C) as well, although in many cases they were carried out on the same C2C infrastructure. Soon after, the amount of transactions carried out under the C2C paradigm started to decrease slightly compared to the amount of transactions performed under the B2C paradigm, which in fact were steadily increasing. This fact, together with the weaknesses of the C2C infrastructure when it is applied to support B2C activities, caused the scientific community to start focusing a great research activity on developing new services, techniques, and methods to improve the user experience and satisfaction in the B2C e-commerce context.

In this paper, we have introduced a set of services provided by intelligent agents and designed in response to the particular necessities of B2C e-commerce users, among which we stress the following ones:

- A dynamical hierarchical cataloguing system to arrange and describe products classes by means of a set of variables. This component allows users to specify their requirements as features of the particular product category to be searched in and to perform comparisons among several products depending on their features.
- A search or product selection system based on fuzzy logic concepts that allows the specification of vague and uncertain descriptions of the required product. This strategy is specially suitable to the case of B2C portals that deal with high involvement products, where the amount of returned relevant results is increased through the use of vague and imprecise search criteria in base of a particular product category features. Besides, the results are clustered into different groups identified by linguistic labels that specify their potential degree of interest to the user.
- A knowledge learning system to help users to learn the existing relationships between the attributes that describe a particular product category. Such knowledge is very useful to deal with the problem of product overestimation, which arises when the users ignore the possible alternatives in the market and which ones are the best suited for their necessities. The inferred knowledge is provided to the user in form of a set of relevant fuzzy association rules. The main advantage of this knowledge representation formalism is that they are self-explicative and highly comprehensible. Besides, they are highly flexible as they allow the definition of different target variables, which allows

to discover the existing relationships between the remainder of variables used to define the search criteria with the selected target variable. Finally, the learned rules are inferred according to the input provided by the user, so they are useful for the user to learn how to tune the search criteria to maximise the relevance of the results.

Finally, we have developed a prototype of a B2C e-commerce portal to demonstrate the benefits of our proposal, which can be accessed online at e-Marketplace (2010). Besides, our development is supported by a multi-agent architecture designed accordingly to the specifications defined by the FIPA (Foundation for Intelligent Physical Agents) committee.

Acknowledgements

This work has been funded by the Regional Government of Castilla-La Mancha under research projects PII2I09-0052-3440 and PII1C09-0137-6488. The authors would also like to thank Raul Miguel Sabariego for the implementation of the e-Zoco B2C portal prototype.

References

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the ACM SIGMOD international conference on management of data* (pp. 207–216).

Albusac, J., López-López, L. M., Murillo, J. M., & Castro-Schez, J. J. (2008). Supporting customer searches in e-marketplaces by means of fuzzy logic-based machine learning. In *Proceedings of The 2008 IEEE/WIC/ACM international conference on intelligent agents technology and web intelligence* (pp. 892–896).

Ashri, R., Rahwan, I., & Luck, M. (2003). Architectures for negotiating agents. *Lecture Notes in Computer Science (Springer), 269*, 136–146.

Cameron, D. (1997). *The new business platform for the Internet*. Charleston: Computer Tecchnology Research Corporation.

Casillas, J., & Martinez-Lopez, F. (2009). Mining uncertain data with multiobjective genetic fuzzy systems to be applied in consumer behaviour modelling. *Expert Systems with Applications (Springer), 362*, 1645–1659.

Casillas, J., Martinez-Lopez, F., & Martinez, F. J. (2004). Fuzzy association rules for estimating consumer behaviour models and its application to explain trust in Internet shopping. *Fuzzy Economic Review, 9*(2), 3–26.

Castro, J. L., Castro-Schez, J. J., & Zurita, J. M. (1999). Learning maximal structure rules in fuzzy logic for knowledge acquisition in expert systems. *Fuzzy Sets and Systems, 101*(3), 331–342.

Castro-Schez, J. J., Jennings, N. R., Luo, X., & Shadbolt, N. R. (2004). Acquiring domain knowledge for negotiating agents: A case of study. *International Journal of Human–Computer Studies, 1*(61), 3–31.

Castro-Schez, J. J., Jimenez, L., Moreno, J., & Rodriguez, L. (2005). Using fuzzy repertory table-based technique for decision support. *Decision Support Systems, 3*(39), 293–307.

Castro-Schez, J. J., Vallejo-Fernández, D., Rodriguez-Benitez, L., & Moreno-García, J. (2008). A fuzzy logic based approach to improve cataloguing and searching in e-commerce portals. *Lecture Notes in Business Information Processing (Springer), 12*, 316–327.

Foundation for Intelligent Physical Agents. <http://www.fipa.org>.

Foundation for Intelligent Physical Agents, FIPA Agent Management Specification (<http://www.fipa.org/specs/fipa00023>).

Foundation for Intelligent Physical Agents, FIPA Agent Message Transport Service Specification (<http://www.fipa.org/specs/fipa00067>).

Geipel, M. M., & Weiss, G. (2007). A generic framework for argumentation-based negotiation. *Lecture Notes in Computer Science (Springer), 4676*, 209–223.

Hindriks, K. V., Jonker, C., & Tykhonov, D. (2008). Towards an open negotiation architecture for heterogeneous agents. In *Proceedings of the 12th international workshop on cooperative information agents* (Vol. XII, pp. 264–279).

Holahan, C., Auctions on eBay: A Dying Breed, 2008. <http://www.businessweek.com/technology/content/jun2008/tc2008062_112762.htm> Retrieved on 16.05.09.

Jonker, C. M., Robu, V., & Treur, J. (2007). An agent architecture for multi-attribute negotiation using incomplete preference information. *Autonomous Agents and Multi-Agent Systems (Springer), 15*, 221–252.

Klaue, S., Kurbel, K., & Loutchko, I. (2001). Automated negotiation on agent-based e-marketplaces: An overview. In *Proceedings of the 13th bled electronic commerce conference* (pp. 508–519).

Kowtha, N. R., & Choon, T. W. I. (2001). Determinants of website development: A study of electronic commerce in Singapore. *Information and Management, 3*(39), 227–242.

Laudon, K. C., & Laudon, J. P. (2005). *Management information systems: Managing the digital firm*. Prentice Hall.

Lee, W. P., Liu, C. H., & Lu, C. C. (2002). Intelligent agent-based systems for personalized recommendations in Internet commerce. *Expert Systems With Applications, 22*(4), 275–284.

Lopez-Lopez, L. M., Miguel, R., Albusac, J., & Castro-Schez, J. J. (2009). Improving searching usability in direct shopping portals. In *Proceedings of The 28th North American fuzzy information processing society annual conference*.

Mamdani, E., & Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man–Machine Studies, 7*(1), 1–13.

Martinez-Lopez, F., & Casillas, J. (2009). Marketing intelligent systems for consumer behaviour modelling by a descriptive induction approach based on genetic fuzzy systems. *Industrial Marketing Management*. doi:10.1016/j.indmarman.2008.02.003.

Mohanty, B. K., & Bhasker, B. (2005). Product classification in the Internet business – a fuzzy approach. *Decision Support Systems, 38*(4), 611–619.

Novak, T., Hoffman, D., & Yung, Y. (2000). Measuring the customer experience in online environments: A structural modelling approach. *Marketing Science, 19*(1), 22–42.

Ryu, Y. U. (1999). A hierarchical constraint satisfaction approach to product selection for electronic shopping support. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 29*(6), 525–532.

Saward, G., & O'Dell, T. (2000). Micro and macro applications of case-based reasoning to feature-based product selection. In *Conference on expert systems*.

Srikumar, K., & Bhasker, B. (2004). Personalized product selection in Internet business. *Journal of Electronic Commerce Research, 5*(4), 216–227.

Srikumar, K., & Bhasker, B. (2004). Personalized recommendations in e-commerce. In *5th world congress on E-business in 25th Mc Master world congress, 2004*.

Turban, E., Lee, J., King, D., & Chung, H. (2000). *Electronic commerce: A managerial perspective*. Prentice Hall.

Vakratsas, D., & Ambler, T. (1999). How advertising works: What do we really know? *Journal of Marketing, 63*(1), 26–43.

Yager, R. R. (2003). Fuzzy logic methods in recommender systems. *Fuzzy Sets and Systems, 136*(2), 133–149.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*(3), 338–353.

Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning-I. *Information Sciences, 8*(3), 199–249.

CubeCart Free & Commercial Online Shopping Cart Solutions. <http://www.cubecart.com/>. Retrieved on 11.06.09.

Miguel, R., & Castro-Schez, J. J., e-Zoco (B|C)2C e-Marketplace Home page. <http://oreto.esi.uclm.es/aplicaciones/ezoco>. Retrieved on 07.09.10.

Joomla! Open Source Content Management System. <http://www.joomla.org/>. Retrieved on 11.06.09.

osCommerce (open source Commerce) e-commerce solutions. <http://www.oscommerce.com/>. Retrieved on 11.06.09.

PrestaShip Free Open-Source e-Commerce Software for the Web 2.0. <http://www.prestashop.com/>. Retrieved on June 11.06.09.

X-Cart Shopping Cart Software & Ecommerce Solutions. <http://www.x-cart.com/>. Retrieved on 11.06.09.

Zen Cart e-commerce shopping cart software. <http://www.zen-cart.com/>. Retrieved on 11.06.09.