Decision Support Systems 81 (2016) 66–75

Contents lists available at ScienceDirect

Decision Support Systems

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

A linguistic mobile Decision Support System based on fuzzy ontology to facilitate knowledge mobilization

J.A. Morente-Molinera, R. Wikström, E. Herrera-Viedma, C. Carlsson *

Dept. of Computer Science and Artificial Intelligence, University of Granada, Spain
IAMSR, Åbo Akademi University, Finland

ARTICLE INFO

Article history:
Received 2 September 2013
Received in revised form 15 June 2015
Accepted 2 September 2015
Available online 7 September 2015

Keywords:
Fuzzy ontology
Group decision making
Android application
Web Platform Application

ABSTRACT

The current development of the Semantic Web has created an increasing demand for methods and systems that can make use of imprecise information. As the amounts of data collected constantly grows, it will not be feasible to overlook imprecise data. We show that a combination of mobile technology and fuzzy ontology with group decision making support methods will facilitate a mobilization of knowledge, offering users a possibility to get decision making support through their mobile devices regardless of the context and location. In this paper, as an illustration and verification, a web platform and an Android application have been developed to help users to choose a suitable wine for different types of dinners.

1. Introduction

Nowadays, users demand more assistance applications to help them with their everyday life. As most users always carry a mobile device with them, this is the artefact they want to get assistance from. Decision support developed for mobile devices is therefore becoming an increasingly important research area. It is also a critical part of knowledge mobilization [1], a movement that will change how knowledge management is conducted. Knowledge mobilization aims at making knowledge obtained from formal research available and usable by every person who is in need of it.

At the same time, developments in the ICT-field have produced a never ending flow of new technical devices that connect to the Internet and allow users to share and consume information regardless of time and location. In order to allow knowledge mobilization to work on these devices, it is necessary implement methods and technologies for them such as Decision Support Systems [2], fuzzy ontologies [3], and recommendation systems [4]. Moreover, all these methods must work together in order to carry out the necessary tasks. Consequently, one of the present challenges is to find ways of connecting methods and technologies that will allow mobile phones to provide real-time knowledge to the user whenever and wherever he/she needs it. In other words, to bring knowledge mobilization to mobile phones. Thanks to server languages such as PHP or JSP [5], database languages such as Oracle and MySQL, and mobile operating systems such as IOS and Android [6], it can be stated that, today, the creation of multiple tools that collaborate as Internet applications is possible.

In this paper, we are going to discuss the implementation of a mobile Decision Support System that gives real time knowledge about a certain topic. By collecting expert knowledge using ontologies [3], it is possible for non-experts to take advantage of expert wisdom and use the advice of experts on topics that should be dealt with. The implemented system uses linguistic modeling in order to ease the way for experts to communicate with the system. It has been repeatedly proven that experts are more comfortable with expressing themselves using words instead of numbers [7]. This is because humans are used to deal with concepts [8]. In the GDM process, consensus measures [9] will be used to help users to reach an agreement. In order to increase clarity and to give an example of a use case, the implemented application deals with the often complex problem of choosing a wine. In it, a set of users must decide which wine they should order depending on their tastes, the food that they have ordered, the price and the context. By combining a Fuzzy Wine Ontology [3,10], group decision making (GDM) support algorithms [7] and the *fuzzyDL* reasoner [11] a Web Platform Application and an Android application have been developed and implemented. Every mobile that has an Internet connection will be able to use the application and users can get access to the knowledge at any time independent of their location and decide on the choice of wine. A GPS or IP location can also be used in order to determine the set of wines that are available at a certain location.

The paper is structured in the following way. First, Preliminaries are presented in Section 2. Section 3 presents the two implementations developed. Section 4 presents a Discussion and Analysis of the implemented applications and Section 5, summarizes some conclusions.

* Corresponding author.
E-mail address: christer.carlsson@abo.fi (C. Carlsson).

http://dx.doi.org/10.1016/j.dss.2015.09.001

2. Preliminaries

To make this paper as self contained as possible, this section presents concepts and definitions that are used later on in the paper.

2.1. GDM and Decision Support Systems

Since the beginning of Decision Support System (DSS) research [12,13] the field has established itself as a broad and innovative area. A Decision Support System consists of three elements: a user interface subsystem, a database subsystem, and a model processing subsystem [14]. With the help of this structure, users are able to manage and analyze data for decision making purposes [15]. The users access the system through an individually tailored user interface, which lately has taken a more web based approach, and where mobile devices give the user the possibility to constantly have access to the system [16].

This development results in knowledge becoming mobile and, simultaneously, that the traditional knowledge management methods will have to adapt to this change [17]. Users are today demanding decision support in all kinds of everyday situations, where everything, from choosing travel routes to what food to eat is assisted by a mobile device [18]. This opens the door for several unexplored opportunities to apply decision support.

Since the 1980s, DSS has numerous times been successfully used for group decision making (GDM) [19,20]. A Group Decision Support System (GDSS) aims at helping a group to reach consensus, whereas the traditional DSSs are aimed at helping an individual decision maker [21].

Taking experts and their knowledge into consideration when making decisions have always been a challenge for decision makers, however, different methods, such as fuzzy sets [22], makes it possible to include also imprecise data into a DSS. For instance, Choudhury et al.[23] included fuzzy preference relations in their approach to handle uncertain factors when reaching consensus in a group.

We feel confident in stating that handling and computing imprecise data and making this knowledge mobile and context adaptive is a very important research topic for the DSS community. Combining this approach with GDM, makes it possible for a group of users to reach consensus, aided by expert advice. GDM is therefore an essential part of the mobile support system that has been implemented, where the GDM algorithms assist users with carrying out the final decision in an efficient way. Examples of these can be seen in [24,25].

2.2. Fuzzy ontology

The emergence of the Semantic Web has revealed that there is a critical need to develop new methods. This concerns several subareas of the Semantic Web, and naturally also ontologies [26]. Current ontology methods have limitations when dealing with imprecise knowledge [27]. Imprecise information is still a crucial part of everyday situations as it is a fast and effective way to handle complexities, but is often overlooked. Examples of imprecise knowledge can be knowledge existing in companies, which is expressed as linguistic information and represents experience and insight collected over some time. If one does not take measures to save this knowledge, it will disappear as employees leave the organization. Storing and enabling a continuous use of this knowledge is a key issue in many organizations [28].

An ontology developed for the Semantic Web has traditionally been based on one of the description logics (DL), which, however, are not suitable for dealing with uncertainties [29]. A possible solution is to utilize a fuzzy ontology. “A fuzzy ontology is simply an ontology which uses fuzzy logic to provide a natural representation of imprecise and vague knowledge and eases reasoning over it” [30]. No unique definition for fuzzy ontology exists, but, the following definition describes a DL-based ontology for the Semantic Web:

Definition 1. [31,32]

A fuzzy ontology is a quintuple $O_F = \{I, C, R, F, A\}$ where: $I = \text{the set of individuals}$, $C = \text{the set of concepts}$, $R = \text{the set of relations}$, $F = \text{the set of fuzzy relations}$, and $A = \text{the set of axioms}$.

Fig. 1 presents the scheme of a fuzzy ontology.

When a query is carried out in the fuzzy ontology, the following steps are followed:

1. *Query providing*: Experts provide a query on the information that they want to obtain from the ontology.
2. *Ontology searching*: Each ontology individual is analyzed in order to determine if it fulfills the query. A previously determined threshold is used and if an individual passes it, it is a desired entity; otherwise, the individual is discarded.
3. *Results presentation*: A ranking of individuals that fulfill the requirements imposed by the users and sorted using their similarity to the query is presented.

Recently there has been an increase in applications based on fuzzy ontologies, for instance, Jiang et al. [33] used modular fuzzy ontologies for management change purposes, and Bobillo and Straccia [34] extended their *fuzzyDL* software [35] with features for handling fuzzy integrals. Fuzzy ontologies are being approved and utilized by the research community.

The Fuzzy Wine Ontology [3,28] was designed to work as a place-holder for applications developed for industrial purposes, and was built with non-classified information. Knowledge about wines, which is naturally imprecise, is a perfect environment for testing Decision Support Systems. The knowledge included in the fuzzy ontology has been collected from websites created by and for wine connoisseurs.[1] Also academic publications, non-academic publications and books have been used to complete the ontology. The measurable wine properties were mostly collected from the Finnish alcohol distribution monopoly Alko (e.g. alcohol level and price). Currently, the ontology contains over 600 wines and is an appropriate testing environment for imprecise expert-based knowledge.

The evaluation of the ontology was carried out by using an application on smart mobile phones to find out how well the application meets its objectives [36]. Brank et al. [37] state that this evaluation approach is a bit vague and has several drawbacks, one being the difficulty to clearly pinpoint how the ontology improves the end result, as the quality and design of the ontology is hard to evaluate. Nevertheless, the results produced from implementing the Fuzzy Wine Ontology with the smart phone application has proved to produce similar answers as professional advice given by wine connoisseurs. For our purposes, we feel that this level of knowledge and expertise is sufficient enough.

The Fuzzy Wine Ontology is composed of the following descriptive attributes:

Country of origin: The location where the wine is produced has a strong impact on the final product. The weather and the different grapes give each wine a special character. This implies that different countries and regions have their own supporters. In the Fuzzy Wine Ontology, four countries are included: France, Spain, Italy, and the USA.

[1] E.g. www.alko.fi, www.winesfromspain.com and www.snooth.com.

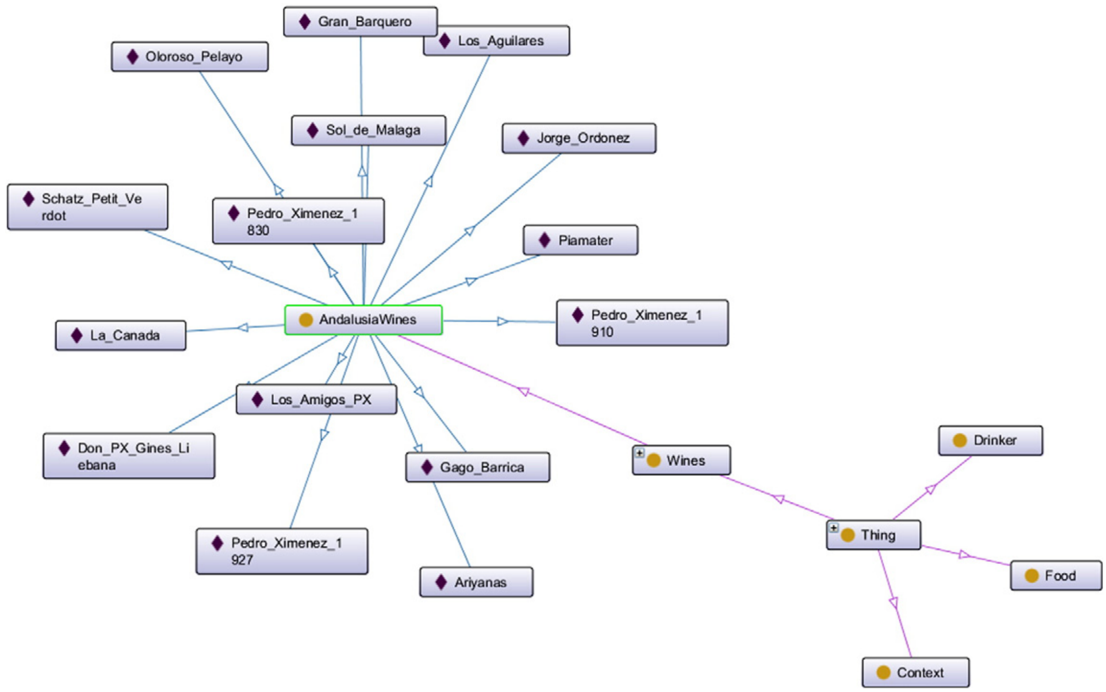

Fig. 1. Fuzzy ontology scheme.

Quality: Wine quality can be judged based on different criteria, such as color, acidity, alcohol, sweetness and body, which all have an impact on the wine taste. Therefore, we include wines of three typical colors (red, white and rosé), four typical wine bodies (light, medium, full and extra full) and also four levels of sweetness (dry, medium dry, medium sweet and sweet).

Context: Depending on the context, wine drinkers select their wines in order to fit a specific dinner. People will alter their wine choice depending on the particular context. The ontology includes 6 different contexts: Formal, Candle, Friends, Business, Family and Picnic. Logically, different contexts demand different attributes from the wines.

Food: Most recommendations for pairing wines are based on the type of food being eaten. Different attributes fit well to different types of food and spices. There are 11 different food categories included: Lamb, Chicken, Beef, Pork, Fish, Game, Salad, Grilled Food, Shellfish, Pasta, Party.

The Fuzzy Wine Ontology is scalable, meaning that one can add and remove wines without affecting the overall functionality. The querying of the ontology works in the following way. First the wine's membership values to the different categories are calculated, then, with the use of the OWA operator [38], the different values and weights are combined to produce a general value that represents the suitability of specific wines for a specific scenario. In this way, the most suitable wines for different contexts can be retrieved.

The Fuzzy Wine Ontology was modeled using the Web Ontology Language (OWL), which offers a family of knowledge representation languages to create ontologies aimed at the Semantic Web. The OWL is supported by the World Wide Web Consortium (W3C) and is a standard language for Semantic Web [39].

The most useful software for OWL ontology creation is, undoubtedly, Protégé [40]. However, the OWL language does not initially support the use of fuzzy sets, which is why the *fuzzyDL* reasoner by [35] was implemented to create the Fuzzy Wine Ontology and the applications developed. *fuzzyDL* extends the DL SHIF and allows the user to define fuzzy concepts. Bobillo and Straccia [11] developed a plug-in that integrates the *fuzzyDL* reasoner with Protégé, to make it possible to implement fuzzy logic in OWL and make fuzzy logic available for general use.

3. A Decision Support System for recommending wine

We combined the Fuzzy Wine Ontology with a decision support algorithm to create a novel Decision Support System that aids dinner guests to choose the most suitable wine for the occasion. Two different versions of the system were developed and implemented:

- *Web platform*: This version was implemented using JavaServer Pages (JSP) and runs over a web browser in any device that has internet access.
- *Android application*: This version consists of an Android app that can be downloaded and installed in any mobile device that supports Android applications.

Both implementations follow the activity diagram [41,42] as shown in Fig. 2. As can be seen, the developed applications build on the following steps:

1. *Location search*: First, the users' location is retrieved using IP location or GPS. This data is sent to Google services in order to retrieve information about the actual location of the used device.
2. *Ontology search*: After location search is performed and search parameters are provided, the Fuzzy Wine Ontology search starts. The parameters for the search are specified as follows:
   (a) *Context*: Refers to the scenario surrounding the dinner. Depending on the purpose of the dinner, some wines can be more suitable than others. Usually one has different criteria for different contexts, e.g. for a formal dinner with important guests, cheap wines can be given a lower importance. Three options are available: Candle, Friends and Formal.
   (b) *Food*: The type of food to be served on the dinner. Depending on this factor, there are some wines that are more suitable than

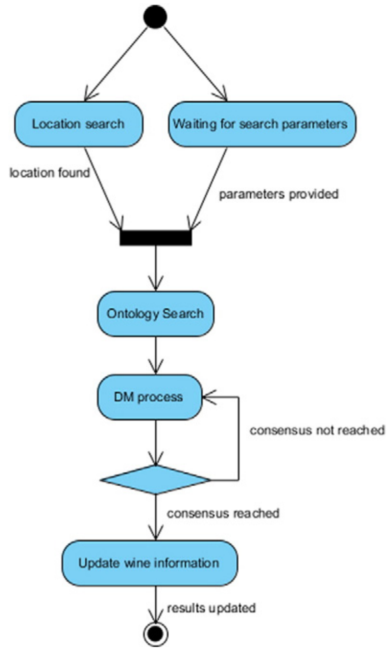

Fig. 2. Web platform and Android application activity diagram.

others. Based on knowledge retrieved from wine experts, different wine properties fit different types of food; it is common wisdom that red wines are more suitable for meat than white wines. Five options are available: Game, Fish, Grilled food, Chicken and Shellfish.

(c) *Number of people*: The number of people that are participating in the dinner party. This parameter will only be used in the group decision making process.
(d) *Number of wines*: The number of wines that the ontology search must provide. The minimum is four (one for each different criterion). This feature allows users to choose how many wines they want to select from in the decision making process.

Because different criteria can be equally valid when a wine is chosen, several searches with different criteria are carried out. Users can choose their favorite wine according to the criteria that best fit them. In total, four searches with four different criteria are done:

(a) *Most famous wine*: This is the most famous wine of the location where the users are. It is selected as the best wine produced at the location or the one that is typically consumed among the natives. This criterion allows users to taste a wine that is characteristic of the place that they are visiting.
(b) *Lowest price wine*: This option retrieves the lowest priced wine from the ontology. It is a suitable criterion for people who are not fond of wines and just want to choose an economic option.
(c) *Best wines according to the context and food*: This option retrieves, using the ontology, a list of the best wines for the context and food specified by the user. The most suitable wine, among the ones available in that location, is chosen.
(d) *Most voted wine*: This criterion takes into account results from previous group decision making processes in order to recommend a specific wine. The most voted wine available in the location is selected and added to the search result list. If no wine from that location has ever been selected by any users, then this criterion is not taken into account.

With these four criteria and the number of wine parameters specified by the users, a list of wines and the reason why they were chosen is presented. The approach follows classical group decision making methods [43–45] and the Decision Support Systems methodology [21, 46].

3. *Decision Making* (*DM*) *process*: Users must decide which wine they want to order among the presented ones. The web platform and Android app implements a group decision algorithm that can assist in the decision.
4. *Updating wine information*: After the group decision making process has ended, the wine–location database is updated adding one to the number of times that the selected wine has been chosen. Thanks to this, posterior decisions will give feedback on what other people have selected on previous occasions. The DM process will be described in more detail in Section 3.1.

Information about which wines are available in each location and how many times a wine has been chosen are stored in a database. Its entity–relation diagram can be seen in Fig. 3. As can be seen, it has two tables and one relationship:

- *Wine table*: This table stores all the wines that conform with the ontology without taking into account their locations and the number of times that the wine has been previously selected.
- *Location table*: This table stores all the locations available. Because this table is independent of the wine, it is possible to add locations, delete them and modify associated wines at any time. Thanks to this database structure, the wine–location association process is dynamic and scalable.
- *Wine–location relationship*: It stores which wines are associated with specific locations. A wine can be associated with multiple locations and in each location there are several wines.

To facilitate the database management, an application has been implemented in order to ease the information updating tasks; wines and locations stored in the wine–location database can be updated at any time.

Incoming requests of mobile devices or web browsers are handled by the server servlet which is in charge of dealing with the ontology API and the wine–location database. When an ontology search is made, the server servlet retrieves the wines that are affiliated with the users' location, and sends the query to the ontology. When the ontology API returns the wine list results, the server servlet sends the resulting information to the mobile device or web browser that has made the request. Both the Web browser and the Android application share the same ontology and wine–location database, that is, the server is the same for both applications. Due to this, decision making results and wine information is shared by the two versions avoiding redundancy issues and easing the information updating task.

In Section 3.1, the used algorithm is explained in more detail and is illustrated with an example. In Sections 3.2 and 3.3, the Web platform and the Android application are described in more detail.

3.1. The implemented GDM algorithm

The implemented GDM algorithm for carrying out the decision making process in the application is based on the one described in [44]. A questionnaire asking the users about their degree of preference among each possible alternative is filled in and preference relation matrices are built using the provided information.

One way of knowing if the users have reached an agreement is to use consensus measures, which makes it possible to calculate the overall agreement among the ranked alternatives. If the consensus is low, it is reasonable to go for another decision making round, but if consensus is high, it means that almost all users agreed, making it useless to repeat the process one more time. Consensus measures can also be used to

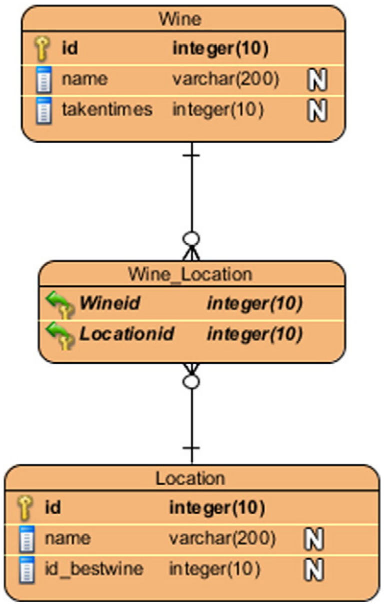

Fig. 3. Wine–location database entity–relation diagram.

advise users of how to modify their opinions in order to reach a higher consensus [9].

The wine selection application that has been created needs human input in order to work. To ease the way for users to express themselves, linguistic modeling has been used [8]. Good overviews of linguistic modeling can be found in [47–49]. Although the research in this field has generated much publications lately, it is also the case that quite a few application papers in other fields claim to use linguistic modeling. For example, in decision support, [45] and [50] use linguistic modeling in order to deal with imprecise information. Linguistic modeling has also been applied satisfactorily to ontologies as can be seen in [51] and [52]. Concretely, in the implemented application, linguistic values belonging to a balanced linguistic term set $S = \{s_1, \ldots, s_n\}$ are used to express the preference degrees. $n = 7$ is considered a number high enough to allow users to express themselves correctly and low enough not to confuse them with unnecessary complexity. When providing labels to the question *How much do you prefer alternative 1 to alternative 2*?, s_1 will indicate that alternative 2 is totally preferred, s_7 will denote that alternative 1 is preferred with the highest possible degree to alternative 2 and s_4 will indicate that they are equally preferred for the user. Using this method, users can communicate with the system in a comfortable way using words. Linguistic modeling is also used for the system to provide recommendations to users on how to modify their opinions in order to reach a consensus. Thanks to linguistic modeling, communication processes become easier to the user who can express himself/herself using methods that are familiar to him/her and also to receive information that will be easy for him/her to understand.

In the GDM algorithm that was implemented for the mobile application, the consensus and alternative ranking values belong to the interval [0,1], therefore, they can be expressed linguistically using a balanced linguistic term set $S = \{s_1, \ldots, s_n\}$ if the following approximation expression is applied [7]:

$$LR_r = s_{(i+1)} | i = \text{round}(r \cdot (n-1)) \tag{1}$$

where r is the numerical value that should be expressed linguistically, LR_r is its linguistic representation, $r \in [0, 1]$ and s_i is a linguistic value that belongs to S.

The GDM process has the following steps:

1. *Providing preferences*: Taking turns, a questionnaire is provided to the users in order to collect their preferences. With the retrieved information, a preference relation matrix is built for each user.
2. *Decision making calculation*: Using the preference matrices, the group decision making algorithm is executed so that a ranking of the selected wines, together with consensus information, is showed to the users.
3. *Preliminary decision making results*: When the results are showed to the users they can decide, using the consensus information, whether to choose the first ranked wine or to modify their preferences. If they choose the second option, the preference providing step is repeated but, this time, advice is supplied to the users in order to make them reach a consensus.
4. *Final result*: When consensus is high enough or users do not want to continue modifying their preferences, the first ranked wine at this stage is chosen and the group decision making process ends.

For a better understanding of the linguistic GDM process used, a brief example of how the algorithm works is presented: Imagine that three dinner guests, e_1, e_2 and e_3 should decide what wine to drink for the dinner. A mobile device is used to search for suitable wines that are available in the restaurant and that fit the purpose of the dinner and the ordered food. After performing the search, the wine alternatives w_1, w_2, w_3 and w_4 are provided to the users. A questionnaire is filled in by the attendants using the balanced linguistic term set S for describing the grade of preference between every two wines.

$$S = \{s_1 : very\_low, s_2 : fairly\_low, s_3 : low, s_4 : medium, s_5 : high, s_6 : fairly\_high, \text{and}\, s_7 : very\_high\}.$$

Using the questionnaire results, a preference relation matrix [44] is built for each dinner guest. Results are shown below:

$$P_1 = \begin{pmatrix} - & s_2 & s_1 & s_3 \\ s_7 & - & s_6 & s_5 \\ s_3 & s_4 & - & s_5 \\ s_1 & s_1 & s_2 & - \end{pmatrix} P_2 = \begin{pmatrix} - & s_3 & s_1 & s_2 \\ s_5 & - & s_7 & s_6 \\ s_4 & s_4 & - & s_3 \\ s_2 & s_1 & s_1 & - \end{pmatrix} P_3 = \begin{pmatrix} - & s_3 & s_1 & s_2 \\ s_5 & - & s_7 & s_6 \\ s_4 & s_4 & - & s_3 \\ s_2 & s_1 & s_1 & - \end{pmatrix}.$$

Aggregating P matrices, the collective preference matrix (C) is calculated [43,44]. Although results are given in the interval [0,6], it is possible to make a domain change and express them in the interval [0,1]. Both matrices are shown below:

$$C_{[0,6]} = \begin{pmatrix} - & 1 & 0 & 1.33 \\ 5.33 & - & 5.66 & 5 \\ 3 & 2.66 & - & 2 \\ 1 & 0 & 0.66 & - \end{pmatrix} C_{[0,1]} = \begin{pmatrix} - & 0.16 & 0 & 0.22 \\ 0.88 & - & 0.94 & 0.83 \\ 0.5 & 0.44 & - & 0.33 \\ 0.16 & 0 & 0.11 & - \end{pmatrix}.$$

Using C, a selection process is carried out. The t-norm *maximum* [22] has been used to compute the final ranking result. The resulting values

Table 1
Results of the selection process for the decision making example.

Alternatives	GDD	GNDD	T (*GDD*, *GNDD*)
w_1	0.1294	0.5927	0.5927
w_2	0.8883	1	1
w_3	0.4255	0.8333	0.8333
w_4	0.0922	0.6294	0.6294

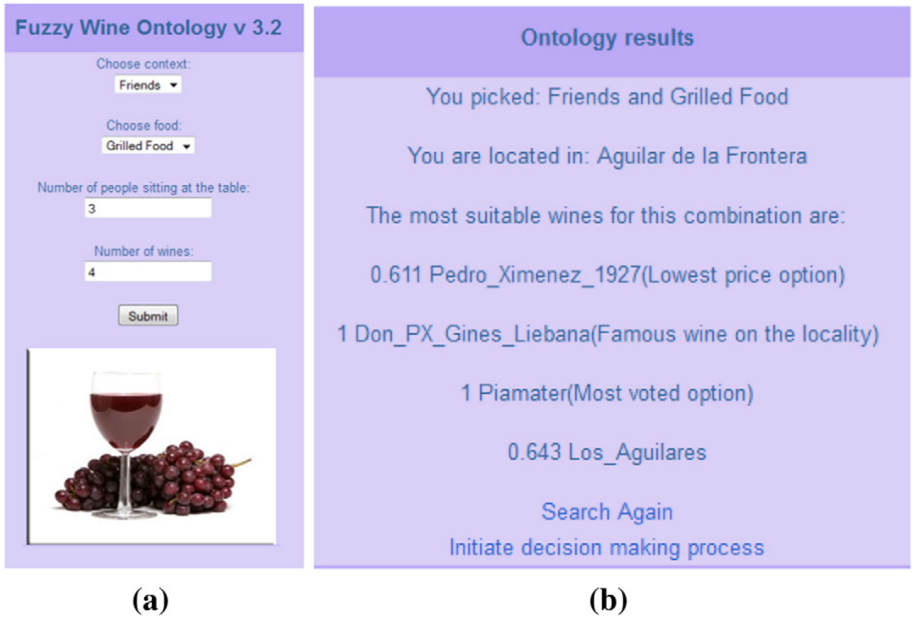

Fig. 4. Web platform, information form screenshot and ontology results screenshot.

for each of the alternatives, belonging to the interval [0,1], are specified in Table 1. According to the table, the final alternative ranking of the group decision making process is $\{w_2, w_3, w_4, w_1\}$; w_2 is the most preferred wine among the dinner guests and w_1 the least preferred option.

In the following, we will work through one of the testing and validation cases we have designed and implemented both as a Web and an Android application. The users are four people in Córdoba, Spain (similar cases have been designed and worked through on several other locations with different groups of users).

3.2. Web Platform Application

The Web Platform Application was created for users whose mobile devices do not have an Android operating system installed, as it can be used in every device that allows an internet connection and has a Web browser installed in it. In the Web platform, the server servlet [5] is the element that handles the communication, presents the results to the user and carries out the group decision making process. In other words, all the computational effort is resolved there. The following software has been used in the web platform implementation:

- The Web platform was implemented using JSP, Javascript and Java languages.
- A Tomcat server is used for running the servlet.
- The wine–location database was built using MYSQL.
- The connection between the server and the database uses JDBC.
- Java Netbeans IDE was the development environment used.

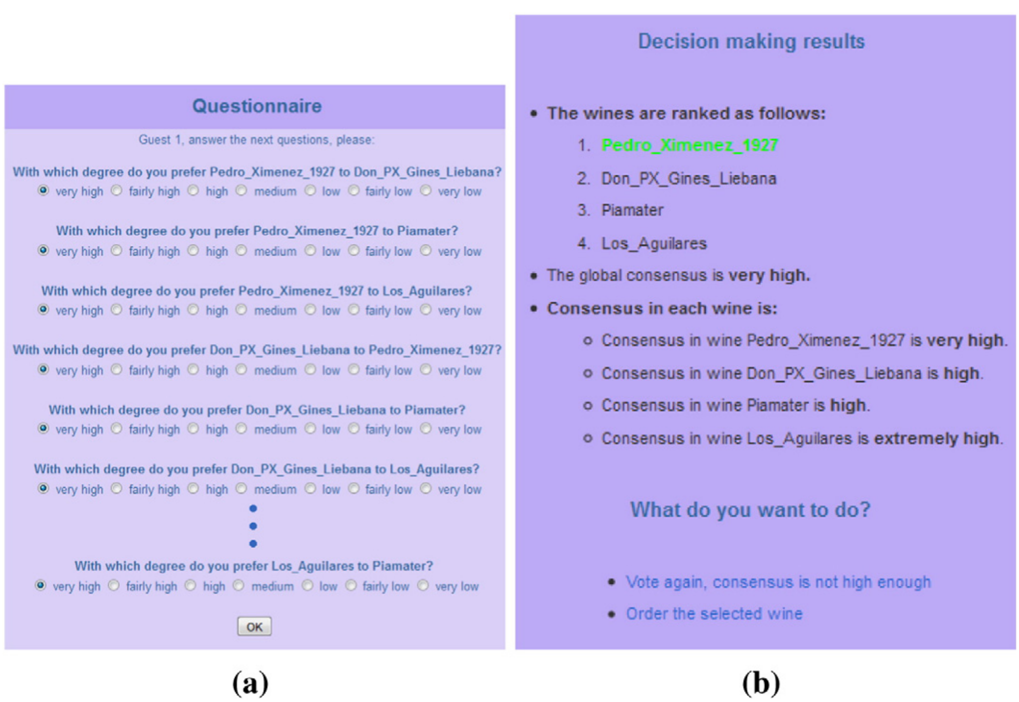

Fig. 5. Web platform, questionnaire screenshot and decision making temporary results screenshot.

Finally, for a better understanding of how the Web Platform Application works, an example is presented:

Four people are seated in a restaurant in Aguilar de la Frontera, a town in Córdoba, Spain. It is an informal meal among friends and they are going to eat grilled food. They want help with finding four wines for them to discuss further about. After filling in all the information on the Web page as Fig. 4a shows, ontology results according to where they are located are shown (Fig. 4b) and the decision making process starts. First, each one of the friends fills in the questionnaire presented in Fig. 5a and, after that, preliminary results are displayed (Fig. 5b). Now, the friends can repeat the decision making process pressing the *vote again* link or select the most voted wine by pressing the *select the most voted wine* link. Because consensus is high, they decide not to go for another decision making round and select the wine: *Pedro_Ximenez_1927*.

3.3. Android application

The Android application follows a client–server model in order to make operations that require a high computational effort to execute in an adequate environment. The three client–server requests that must be performed in order to complete the process are described in more detail below:

1. *Location request*: Google servers are used to retrieve the mobile device location. An IP address or GPS coordinates can be used for this purpose; it is up to the user to decide which method to use. Furthermore, mobile devices that have an Android operating system but do not have a GPS component can still use the Android version.
2. *Ontology search result request*: Fuzzy ontology searches are computationally intensive and cannot be executed on the mobile platform. Because of that, search data is sent to a server that makes the ontology search and returns the results.
3. *Update the wine–location database with decision making results*: The mobile device sends the final decision making results to a servlet applet that updates the wine–location database. Because the database contains overall information of all the decision making processes carried out by all the devices that have used the app, it has to be stored on the server in order to enable the devices to use the same database. This way, redundancy is avoided and wine–location assignments can be changed, without having to update the application for all the mobile devices separately, modifying only one database.

The following software were used in the Android implementation:

- The same server is used and the servlet programming languages are the same as in the web platform version: JSP, Javascript and Java.
- Sockets are used in the Android application–server communication for the ontology search information sharing.
- Java language was used for programming the Android application.
- For security reasons, the connection between the Android application and the database is made through the server, not directly via a JSP script.
- Eclipse IDE and the Software Development Kit that Android provides were the development environments used.

Finally, for a better understanding of how the Android application works, screenshots of the example presented in the previous section is shown. Fig. 6a shows the meal information input part of the Android application and in Fig. 6b, the ontology result screen is displayed. Fig. 7a shows an example of a poll question. Questions are shown one by one to the dinner guests for better readability. Fig. 7b presents an example of the preliminary decision screen.

4. Discussion

A novel application that combines fuzzy ontology with a decision support algorithm has been developed and implemented. The goal is to create a Decision Support System that helps users to choose the wine that best fits them for various types of food in different dinner contexts.

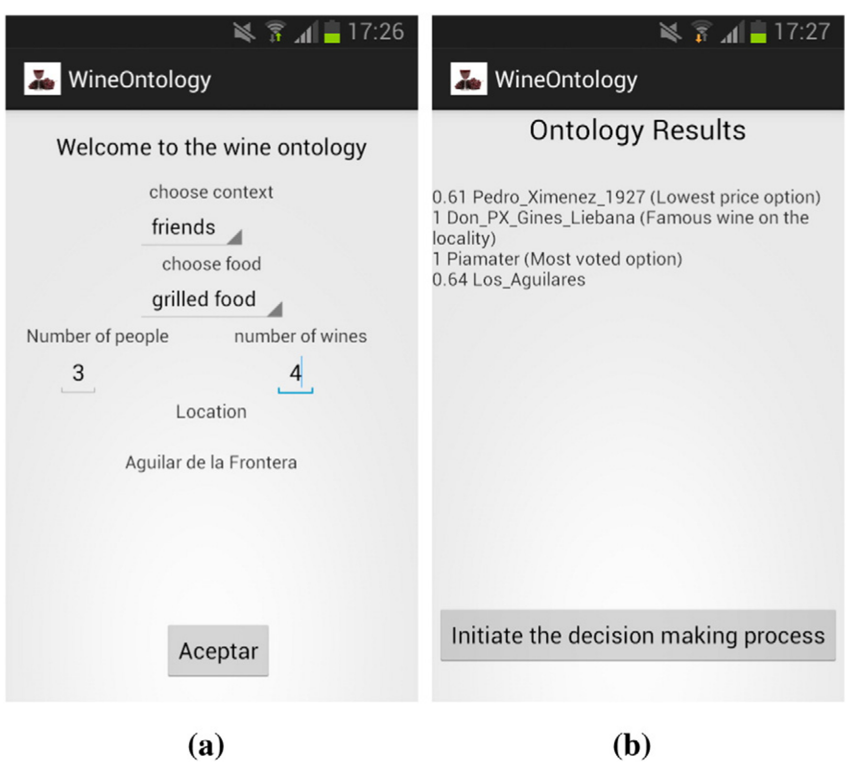

Fig. 6. Android application, search information screenshot and wine ontology results screenshot.

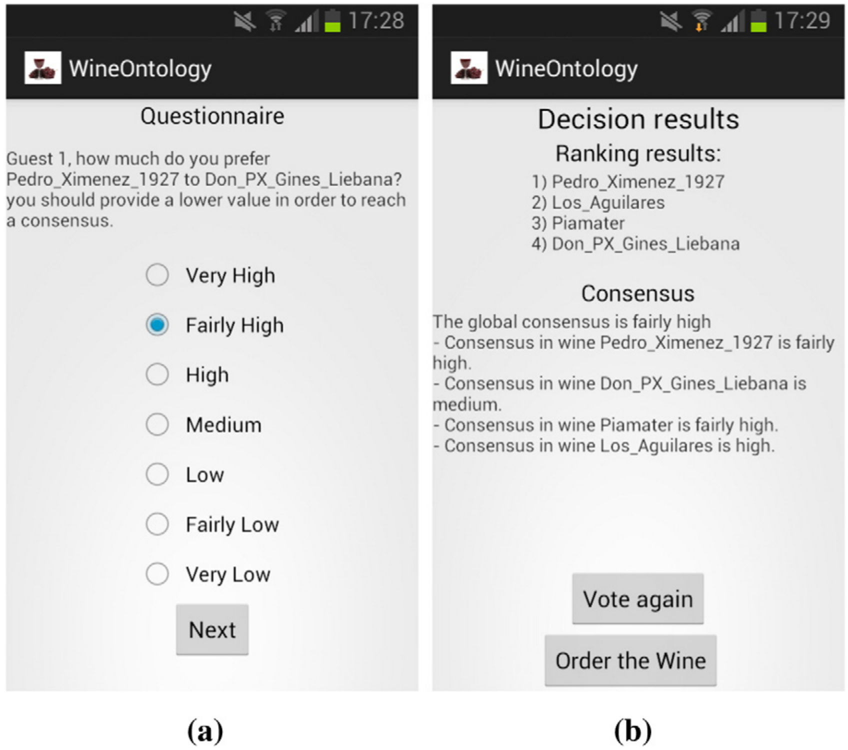

Fig. 7. Android application, questionnaire screenshot and temporary results decision screenshot.

Thanks to the fuzzy ontology, the knowledge of wine connoisseurs is often offered in an imprecise, linguistic form available for the application users to benefit from. Dinner guests that do not know too much about wines can get support and make a good wine selection. Decision support algorithms [21,46] provide users with an accurate method to make a decision. Users can discuss and vote for their favorite wines in an efficient and organized way. Consensus measures [25,50] give them a clear overview of how the other dinner guests have voted and help them to reach an agreement from the advice given.

We have found that The Fuzzy Wine Ontology is a valid ontology due to the OWL constructs and reasonably complete as most of their characteristics that are used by sommeliers to describe and classify a wine are taken into account as concepts. Also, every individual entity is related to every concept avoiding missing information. Due to the fact that our ontology is constantly updated, its coverage will increase with every wine added. Now, the most important wines of every European country are included. Data has been retrieved from well-known web-sites with expert knowledge of wines. Because the set of entities of the ontology and the set of concepts are not interacting, an application based evaluation is sufficient to test the correctness of the ontology [37]. If the data included is correct, then it can be stated that the ontology is valid [36].

GPS and/or IP location features are used in order to retrieve the users' location. Therefore, wine recommendations are location dependent, that is, the wine list provided to the guests contains only wines that are available in their actual location. Wines that are not available are omitted in order to avoid impossible choices and to speed up the computations.

The application has been designed to be used in mobile devices. Dinner guests can use it and make decisions in real time at the restaurant where they are going to have dinner. For users that do not have Android installed in their mobile devices, a Web Platform version has been implemented that can be used by any mobile device with an Internet connection. Although the wine selection problem has been the approach used in this paper, this decision making support scheme can be applied to assist users in a number of other situations. For example:

- Information about loans from banks located in a specific location could be stored in an ontology in order to help users to select the loan that is most adequate for their current situation.
- Data about apartments available for rent or sales in a specific location could be used to advise users about the ones that best fit them.
- If travel information is stored in an ontology, this system can be used to advise a group of friends about where they should go on holidays.
- In companies, if experts' knowledge about company management is stored in an ontology, a Decision Support System to advise non-expert members of the company how to make certain critical decisions can be built.
- For investors, this approach can help them to choose where to invest their money in order to obtain the highest benefit.

Apart from applying the same scheme to other fields, there are other future upgrades that can be used to improve the system. The created wine selection Decision Support System allows users to select one wine. An interesting approach would be if the application could help them to select one wine for each dish of the dinner.

Methods to increase the speed of the application should be investigated and applied. The speed of the application is directly dependent on the available wines in a specific location. As more wines are included the ontology search time increases dramatically. Because the application has real time requirements, a solution should be found in order to allow it to search among a large number of items.

The actual application searches for wines according to the locality where the dinner guests are having the meal. Using GPS coordinates to get a more precise address, it could be possible to make the ontology search dependent on the wines available in the restaurant instead of the whole locality.

5. Conclusions

In this paper, web platform and Android applications for selecting a wine at a dinner party have been constructed and tested. Both applications employ a Fuzzy Wine Ontology as their main source of knowledge, making it possible to manage imprecise information in the process. These applications show that it is possible to offer good decision support with mobile devices. This is a good example of how knowledge mobilization can help people to process data and take advantage of it.

In the context of wines, information is imprecise since it comes from the opinions of wine connoisseurs; opinions are imprecise due to their linguistic and conceptual nature. Without fuzzy ontology to model and manage imprecise knowledge, it would not be possible to capture the advice, judgmnt and opinions of sommeliers and wine connoisseurs of various kinds.

The main contribution in this paper is that we show that expert knowledge, imprecise in nature, traditionally stored and analyzed in non-mobile devices, can be activated and successfully managed using mobile devices. The data retrieved by mobile devices can be combined with group decision making algorithms to create useful applications that allow a group of users to reach consensus [25,50]. This mobilization of knowledge [1,53] will make it possible for users to get knowledge support for their decisions, based on imprecise data, regardless of where they are as long as Internet is available.

We present an example of how to make the most of imprecise data that is available in information systems and to combine decision support algorithms with fuzzy ontology. The used structure can be applied to resolve many other situations apart from selecting wines for dinner. For example, if imprecise information and opinions about apartments in a city is available, it is possible to build a similar application to aid people with choosing the apartment that best fits them.

In the future, it is assumed that mobile devices will become increasingly more adopted and used in different decision making contexts, solving tasks that were impossible to manage a couple of years ago. This movement will change the way decisions are made in our everyday life, supporting informed and rational decision making regardless of where we are. In the future, we aim to apply the presented approach in other contexts as well as improving its general usability.

Acknowledgments

This paper has been developed with the financing of FEDER funds in TIN2013-40658-P and Andalusian Excellence Project TIC-5991.

References

[1] H.L. Gainforth, A.E. Latimer-Cheung, P. Athanasopoulos, S. Moore, K.A.M. Ginis, The role of interpersonal communication in the process of knowledge mobilization within a community-based organization: a network analysis, Implementation Science 9 (2014) 59.
[2] H. Demirkan, D. Delen, Leveraging the capabilities of service-oriented decision support systems: putting analytics and big data in cloud, Decision Support Systems 55 (2013) 412–421.
[3] C. Carlsson, J. Mezei, M. Brunelli, Fuzzy ontology used for knowledge mobilization, International Journal of Intelligent Systems 28 (2012) 52–71.
[4] Á. Tejeda-Lorente, C. Porcel, E. Peis, R. Sanz, E. Herrera-Viedma, A quality based recommender system to disseminate information in a university digital library, Information Sciences 261 (2014) 52–69.
[5] M. Hall, More Servlets and JavaServer Pages, Prentice Hall PTR, 2001.
[6] A. Developers, What is Android? 2011.
[7] F. Herrera, S. Alonso, F. Chiclana, E. Herrera-Viedma, Computing with words in decision making: foundations, trends and prospects, Fuzzy Optimization and Decision Making 8 (2009) 337–364.
[8] L. A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning, Information Sciences, Part I, II, III 8,8,9 (1975) 199–249,301–357,43–80.
[9] F.J. Cabrerizo, J.M. Moreno, I.J. Pérez, E. Herrera-Viedma, Analyzing consensus approaches in fuzzy group decision making: advantages and drawbacks, Soft Computing 14 (2010) 451–463.
[10] R. Wikström, Ontology of imprecision and fuzzy ontology applications, in: K.-H. Krempels, A. Stocker (Eds.),Proceedings of the 9th International Conference on Web Information Systems and Technologies (WEBIST 2013), ISBN: 978-989-8565-54-9 2013, pp. 284–287.
[11] F. Bobillo, U. Straccia, Fuzzy ontology representation using OWL 2, International Journal of Approximate Reasoning 52 (2011) 1073–1094.
[12] H.A. Simon, The New Science of Management Decision, Harper & Brothers, 1960.
[13] R.N. Anthony, Planning and Control Systems: A Framework for Analysis, Division of Research, Graduate School of Business Administration, Harvard University, Boston, MA, 1965.
[14] S. Haag, M. Cummings, J. Dawkins, Management information systems, Multimedia Systems 279 (1998) 280–297.
[15] A.A. Angehrn, T. Jelassi, DSS research and practice in perspective, Decision Support Systems 12 (1994) 267–275.
[16] J. Cowie, F. Burstein, Quality of data model for supporting mobile decision making, Decision Support Systems 43 (2007) 1675–1683 (Special Issue Clusters).
[17] J. Shim, M. Warkentin, J.F. Courtney, D.J. Power, R. Sharda, C. Carlsson, Past, present, and future of decision support technology, Decision Support Systems 33 (2002) 111–126 (Decision Support System: Directions for the Nest Decade).
[18] H. van der Heijden, Mobile decision support for in-store purchase decisions, Decision Support Systems 42 (2006) 656–663.
[19] J.-M. Blin, M.A. Satterthwaite, Individual decisions and group decisions: the fundamental differences, Journal of Public Economics 10 (1978) 247–267.
[20] A. Lewandowski, SCDAS — decision support system for group decision making: decision theoretic framework, Decision Support Systems 5 (1989) 403–423.
[21] P. Gray, Group decision support systems, Decision Support Systems 3 (1987) 233–242.
[22] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.
[23] A. Choudhury, R. Shankar, M. Tiwari, Consensus-based intelligent group decision-making model for the selection of advanced technology, Decision Support Systems 42 (2006) 1776–1799.
[24] R.M. O'Keefe, T. McEachern, Web-based customer decision support systems, Communications of the ACM 41 (1998) 71–78.
[25] I. Palomares, R.M. Rodrguez, L. Martnez, An attitude-driven web consensus support system for heterogeneous group decision making, Expert Systems with Applications 40 (2013) 139–149.
[26] V. Nebot, R. Berlanga, Building data warehouses with semantic web data, Decision Support Systems 52 (2012) 853–868 (1)Decision Support Systems for Logistics and Supply Chain Management 2)Business Intelligence and the Web).
[27] F. Bobillo, J. Gómez-Romero, P.L. Araúz, Fuzzy ontologies for specialized knowledge representation in WordNet, Advances on Computational Intelligence, Springer 2012, pp. 430–439.
[28] C. Carlsson, M. Brunelli, J. Mezei, Decision making with a fuzzy ontology, soft computing — a fusion of foundations, Methodologies and Applications 16 (2012) 1143–1152.
[29] T. Lukasiewicz, U. Straccia, Managing uncertainty and vagueness in description logics for the semantic web, Web Semantics: Science, Services and Agents on the World Wide Web 6 (2008) 291–308 (Semantic Web Challenge 2006/2007).
[30] F. Bobillo, Managing Vagueness in OntologiesPh.D. thesis University of Granada, Spain, 2008.
[31] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider, The Description Logic Handbook: Theory, Implementation and Applications, Cambridge University Press, 2003.
[32] S. Calegari, D. Ciucci, Fuzzy ontology, fuzzy description logics and fuzzy-owl, Applications of Fuzzy Sets Theory 4578 (2007) 118–126.
[33] Y. Jiang, Y. Tang, Q. Chen, J. Wang, Reasoning and change management in modular fuzzy ontologies, Expert Systems with Applications 38 (2011) 13975–13986.
[34] F. Bobillo, U. Straccia, Aggregation operators for fuzzy ontologies, Applied Soft Computing 13 (2013) 3816–3830.
[35] F. Bobillo, U. Straccia, fuzzydl: an expressive fuzzy description logic reasoner, FUZZ-IEEE, 2008 923–930.
[36] P.D. Haghighi, F. Burstein, A. Zaslavsky, P. Arbon, Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings, Decision Support Systems 54 (2012) 1192–1204.
[37] J. Brank, M. Grobelnik, D. Mladenić, A survey of ontology evaluation techniques, Proceedings of Conference on Data Mining and Data Warehouses (SiKDD 2005), 2005.
[38] R.R. Yager, On ordered weighted averaging aggregation operators in multicriteria decision making, IEEE Transactions on Systems, Man, and Cybernetics 18 (1988) 183–190.
[39] M. Horridge, M. Krötzsch, B. Parsia, P. Patel-Schneider, S. Rudolph, OWL 2 Web Ontology Language, Primer, W3C Working Group, 2009.
[40] H. Knublauch, R. Fergerson, N. Noy, M. Musen, The Protégé OWL plugin: an open development environment for semantic web applications, The Semantic Web — ISWC 20042004 229–243.
[41] M. Chen, X. Qiu, W. Xu, L. Wang, J. Zhao, X. Li, UML activity diagram-based automatic test case generation for Java programs, The Computer Journal 52 (2009) 545–556.
[42] M. Dumas, A.H. Ter Hofstede, UML activity diagrams as a workflow specification language, UML 2001—The Unified Modeling Language. Modeling Languages, Concepts, and Tools, Springer 2001, pp. 76–90.

[43] J. Kacprzyk, Group decision making with a fuzzy linguistic majority, Fuzzy Sets and Systems 18 (1986) 105–118.
[44] I.J. Pérez, F.J. Cabrerizo, E. Herrera-Viedma, A mobile Decision Support System for dynamic group decision-making problems, IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans 40 (2010) 1244–1256.
[45] J. Lan, Q. Sun, Q. Chen, Z. Wang, Group decision making based on induced uncertain linguistic OWA operators, Decision Support Systems 55 (2013) 296–303.
[46] P.G. Keen, M.S.S. Morton, Decision Support Systems: An Organizational Perspective, volume 35, Addison-Wesley, Reading, MA, 1978.
[47] M. Reformat, C. Ly, Ontological approach to development of computing with words based systems, International Journal of Approximate Reasoning 50 (2009) 72–91.
[48] Y. Tang, J. Lawry, Linguistic modelling and information coarsening based on prototype theory and label semantics, International Journal of Approximate Reasoning 50 (2009) 1177–1198.
[49] J. Morente-Molinera, I. Perez, M. Ureña, E. Herrera-Viedma, On multi-granular fuzzy linguistic modeling in group decision making problems: a systematic review and future trends, Knowledge-Based Systems 74 (2015) 49–60.
[50] S. Alonso, I.J. Pérez, F.J. Cabrerizo, E. Herrera-Viedma, A linguistic consensus model for web 2.0 communities, Applied Soft Computing 13 (2013) 149–157.
[51] J.A. Rodger, A fuzzy linguistic ontology payoff method for aerospace real options valuation, Expert Systems with Applications 40 (2012) 2828–2840.
[52] J.A. Bateman, J. Hois, R. Ross, T. Tenbrink, A linguistic ontology of space for natural language processing, Artificial Intelligence 174 (2010) 1027–1071.
[53] A. Jashapara, W.-C. Tai, Knowledge mobilization through e-learning systems: understanding the mediating roles of self-efficacy and anxiety on perceptions of ease of use, Information Systems Management 28 (2011) 71–83.

Juan Antonio Morente-Molinera is a PhD student in the department of Computer Science and Artificial Intelligence at the University of Granada. He is a member of the Soft Computing and Intelligence Information Systems research group. He holds an MSc in Soft Computing and Intelligence Systems and another one in High School, Language and Professional Training Teaching. His research interests are in group decision making, decision support systems, consensus models, linguistic modelling, fuzzy ontologies and information aggregation.

Dr Robin Wikström is currently a postdoc researcher in the Laboratory of Industrial Management at Åbo Akademi University in Finland. In his doctoral thesis in Information Systems at IAMSR he worked on decision support systems, fuzzy ontologies and knowledge mobilisation in research projects carried out in cooperation with major Finnish multinational corporations. His post-doc research topics represent a continuation of his thesis work, concentrating on facilitating decision making and encouraging decisions makers to think.

Professor Enrique Herrera-Viedma is Professor of Computer Science and the Vice-President of Research and Innovation in University of Granada. He received the M.Sc. and Ph.D. degrees in Computer Science from the University of Granada in 1993 and 1996, respectively. His h-index is 48 with more than 7500 citations received [WoS]. He was identified in 2014 and 2015 as one of the world's most influential researchers by the Shanghai Center and Thomson Reuters in both Computer Science and Engineering. His current research interests include group decision making, consensus models, linguistic modelling, and aggregation of information, information retrieval, bibliometric, digital libraries, web quality evaluation, recommender systems, and social media.

Professor Christer Carlsson is the Research Director of the Institute for Advanced Management Systems Research [IAMSR] at Abo Akademi University and the Past President of the International Fuzzy Systems Association. He received his DSc (BA) degree from Abo Akademi University in Management Science in 1977. His h-index is 34 with more than 7000 citations [WoS]. He received the VII Kaufmann Prize and Gold Medal of SIGEF for research in uncertain management and economy, the Nystrom Prize for scientific excellence from Societas Scientarium Fennia and he is an IFSA Fellow and an IEEE Senior Member. At IAMSR he has worked on research projects in partnership with major Finnish multinational corporations; his research started with mathematical systems theory and operational research and then moved on to multi-criteria optimization, decision support systems, fuzzy optimization, foresight, soft computing, mobile value services, knowledge mobilization, fuzzy real options valuation and fuzzy ontology.