Using Fuzzy Data Mining for finding preferences in adventure tourism

A. Salguero	F. Araque	R. Carrasco	A. de Campos	L. Martínez
Dpto. LSI	Dpto. LSI	Dpto. LSI	E.T.S.I.I.T.	Dpto. de Informática
Univ. de Granada	Univ. de Granada	Univ. de Granada	Univ. de Granada	Univ. de Jaén
Andalucía (Spain)	Andalucía (Spain)	Andalucía (Spain)	Andalucía (Spain)	Andalucía (Spain)
agsh@ugr.es	faraque@ugr.es	faraque@ugr.es	adcr@ugr.es	martin@ujaen.es

Abstract

Soaring is a recreational activity and competitive sport where individuals fly un-powered aircrafts known as gliders. Soaring site selection process depends on a number of factors, resulting in a complex decision-making task. It is common for the decision makers to use their subjective judgment and previous experience when selecting the most appropriate place for soaring. In this paper we address the problem of finding and validating the hidden, subjective preferences which arise in the site selection process for soaring. We propose a data mining process using fuzzy logic (Fuzzy SQL language) to discover the knowledge that the travel agencies or tour operators specialized in adventure tourism needs to make decisions about the more suitable activities suggest to their customers. We provide an example showing how to validate an assessment about a customer's preference related to the temperature evolution and the quality of a day for soaring.

Keywords: Tourism management, DSS, Fuzzy Logic, Data Mining, Functional Dependecies

1 Introduction

As pointed out by Lexhagen [9], tourism businesses should try to develop more value-added services aimed to support the customer in the post-consumption phase. The goal is to build up strong customer relationships and loyalties, which may provide continuous buying behavior. Some examples of ICT (Information and Comunication Technology) value-added services that a tourism enterprise can offer are automatic categorization of user travel preferences in order to match them up with travel options [5] or a search engine interface metaphors for trip planning [12]. A DSS (Decision Support System) for adventure practice recommendation can be offered as a post-consumption value-added service by travel agencies to their customers. Therefore, once a customer makes an on-line reservation, the travel agency can offer advice about adventure practices available in the area that customer may be interested in. Due to the high risk factor accompanying most adventure sports, a regular information system is far from being accurate. A more sophisticated ICT system is required in order to extract and process quality information from different sources. In this way, the customer can be provided with true helpful assistance to be aided in the decision-making process.

Soaring is a recreational activity and competitive sport where individuals fly un-powered aircrafts known as gliders. The soaring community is very extensive. These pilots have had to sharpen up their good meteorological sense to maximize their soaring experience. In order to provide information for predicting patterns and trends more convincingly and for analyzing a problem or situation more efficiently, an integrated DSS designed for this particular purpose is needed.

Prior to the development of this DSS, we need to a find out what are the parameters mainly involved in the decision-making process. Site selection process depends on a number of factors, resulting in a complex decision-making task. It is common for the decision makers to use their subjective judgment and previous experience when selecting the most appropriate place for soaring. To solve this problem we ask to experts in soaring to score a list of possible situation. Instead of using linguistic labels, we ask to experts to give a number reflecting the quality of the day weather conditions. The final score are calculated taking into account factors like temperature, humidity, wind direction or strength and their evolution over the day.

Tour-operators and travel agencies can use the DSS in order to foretell (taking into consideration both former flights data loaded in the Data Warehouse (DW) [1] and the one week weather forecast) whether the conditions for soaring will be favorable and in which geographical areas they could be performed (in our example the data correspond to Granada and its province). In this way, trips can be arranged and activities can be organized with great reliability.

The use of Data Mining (DM) processes will help us to find out the patterns, features and in general the knowledge we are looking for. In fact to find out the features, patterns, etc. we have used Functional Dependencies (FD) and Gradual Dependencies (GRD) [8] because they reflect immutable properties in a DB hence to discover the knowledge we want to.

FD correspond to correlations among data items and are expressed in rule form showing attribute-value conditions that commonly occur at the same time in some set of data. In the regular case, a functional dependency, denoted by $X \rightarrow Y$, expresses that a function exists between the two sets of attributes X and Y, and it can be stated as follows: for any pair of tuples t1 and t2, if t1 and t2 have an equal value on X, they also have the same value on Y. Another way of considering the connections between data in databases is to specify a relationship between objects in a dataset and reflect monotonicity in the data by means of that we have called as GRDs. GRD is a concept closely related to the idea of gradual rules introduced by Dubois and Prade [3].

In this paper we propose to develop a DM process based on the fuzzy logic in order to make it more flexible. To do so, we relax the concept of FD and GRD by means of Fuzzy FD (FFD) and Fuzzy GRD (FGRD) that are quite suitable to model non immutable properties existing in the current manifestation of the data.

This paper is organized as follows: section 2 introduces different preliminaries that are necessary to understand the proposal. In Section 3 is introduced the definition of Fuzzy Global Dependencies (FGDs) based on the FSQL operators. In Section 4 a DM process are showed for finding FGDs (normal ED or ED with a degree of confidence). In section 5 are presented some experimental results and the paper is concluded in section 6.

2 Preliminaries

2.1 Related work

In the last decade many decision-making system which have to deal with multi-criteria decision problems and qualitative information have shown the capability of Fuzzy Decision Analysis (FDA). Liang and Wang [10] proposed the FDA, which uses fuzzy set representations and utilizes linguistic variables for rating qualitative factors to aggregate decision-makers' assessments, and applied it on facility site selection and personnel selection. Ghotb and Warren [4] employed FDA to evaluate the necessity of adopting a new hospital information system.

On the other hand, the problem of FD inference has been treated many times in literature. Mannila and Räihä [6] proposed a heuristic algorithm for finding functional dependencies. Akutsu and Takasu [11] studied inference of functional dependencies from data with small noise, and gave PAC-type analyses. Investigated for long years, this issue has been recently addressed in a novel and more efficient way by applying principles of data mining algorithms. In this case, the inference of FD is carried out analyzing the data stored in a data base. This method is useful when we have large sets of materialized data (e. g. DW environments…).

The concept of FFD given by Cubero and Vila in [8] is a smoothed version of the classical FD. The basic idea consists in replacing the equality used in the FD definition by fuzzy resemblance relations. We can obtain a fuzzy version of GRD (FGRD) in a similar way. We call Fuzzy Global Dependencies (FGD) to the integration of both FFD and FGRD.

The main advantage of FGDs is that they allow us to infer more knowledge from data. Using regular dependencies, only rules that are fulfilled by all of the instances are valid. Using FGDs we can discover dependencies although there are instances that do not fulfil them completely. Furthermore, we can obtain the fulfilment degree for each FGD stated.

The DM process proposed will obtain FGDs by using a flexible query language as the Fuzzy SQL (FSQL) [7], which will provide the information that will support travel agencies decisions about which type of activities are more suitable to do given the specific weather conditions and clients' characteristics.

2.2 FSQL: A language for flexible queries

We have developed a language (FSQL) to manage uncertainties and imprecise information [2]. We have extended the SQL language to allow flexible queries. Thus, the language can manage fuzzy attributes, from different nature that is necessary in our problem, which are classified by the system in 3 types:

- Type 1: These attributes are totally crisp, but they have some linguistic trapezoidal labels defined on them.

- Type 2: These attributes admit crisp data as well as possibility distributions over an ordered underlying domain.

- Type 3: On these attributes, some labels are defined and on these labels, a similarity relation has yet to be defined. These attributes have no relation of order.

The Fuzzy Meta-knowledge Base (FMB) stores information for the fuzzy treatment of the fuzzy attributes in order to define:

- Representation Functions: these functions are used to show the fuzzy attributes in a comprehensible way for the user and not in the internally used format.

- Fuzzy Comparison Functions: they are utilized to compare the fuzzy values and to calculate the compatibility degrees (CDEG function)

We have extended the SELECT command to express flexible queries and, due to its complex format, we only show an abstract with the main extensions added to this command:

- Fuzzy Comparators: In addition to the common comparators (=, >, etc), FSQL includes fuzzy comparators of two trapezoidal possibility distributions A, B with A=$[αA,βA,γA,δA] B=$[αB,βB,γB,δB]. In the same way as in SQL, fuzzy comparators can compare one column with one constant or two columns of the same type. Necessity comparators are more restrictive than possibility comparators, i.e. their fulfillment degree is always lower than the fulfillment degree of their corresponding possibility comparator. More information can be found in [7].

- Fulfillment Thresholds γ: For each simple condition a Fulfillment threshold may be established with the format <condition> THOLDγ, indicating that the condition must be satisfied with a minimum degree γ in [0, 1] fulfilled.

- CDEG(<attribute>) function: This function shows a column with the Fulfillment degree of the condition of the query for a specific attribute, which is expressed in brackets as the argument.

- Fuzzy Constants: In FSQL we can use a set of fuzzy constants.

- Fuzzy Quantifiers: They can either be relative or absolute with the formats $Quantifier [FUZZY] (<condition>) THOLD χ or $Quantifier [FUZZY] (<condition_1>) ARE (<condition_2>) THOLD χ, indicating that the quantifier must be satisfied with a minimum degree χ in [0,1] fulfilled.

We have a FSQL Server available to obtain the answers to FSQL queries for Oracle© DBMS. The FSQL Server maintains a Fuzzy Meta-knowledge Base (FMB) which has all the information about the attributes susceptible to fuzzy treatment.

3 Fuzzy functional dependencies and gradual functional dependencies

There have been several approaches to the problem of defining the concept of FFD but unlike classical FDs one single approach has not dominated. We begin by briefly describing the concept of classical FD, later we give a general definition of FFD and GRFD based on fuzzy functions and then, we shall introduce a more relaxed definition of FFD and GRFD in order to manage exceptions.

The relation *R* with attribute sets $X=(x_1,\ldots,x_n)$, and $Y=(y_1,\ldots,y_m)$ in its scheme verifies the **FD X→Y** if and only if, for every instance *r* of *R* it is verified:

$$\forall t_1, t_2 \in r,\ t_1[X] = t_2[X] \Rightarrow t_1[Y] = t_2[Y]$$

The basic idea of FFDs consists in replacing the equality used in the FD definition by fuzzy resemblance relations, in such a way that: The relation *R* verifies an α–ß FFD $X \rightarrow_{FT} Y$ if and only if, for every instance *r* of *R* it is verified:

$$\forall t_1, t_2 \in r,\ F(t_1[X], t_2[X]) \geq \alpha \Rightarrow T(t_1[Y], t_2[Y]) \geq ß$$

where F and T are fuzzy resemblance relations.

The flexibility provided by the combined use of the parameters α and ß and the different kinds of resemblance relation should be noted. If F is a weak resemblance measure and T is a strong one, we get interesting properties for database design (decomposition of relations). A more detailed description of these concepts can be found in [8].

Often just a few tuples in a database can prevent the FFD from being completed. To avoid this, we can relax the FFD definition in such a way that all the tuples of the relationship are not forced to fulfill the above condition, therefore we define:

Definition 1 (confidence of a FFD). The relation *R* verifies an α–ß FFD $X \rightarrow_{FT} Y$ with confidence *c*, where *c* is defined as:

$$c = 0 \text{ if } Card\{(t_1, t_2)\ t_1, t_2 \in r \,/\, F(t_1[X], t_2[X]) \geq \alpha\} = 0$$

$$c = \frac{Card\{(t_1, t_2)\ t_1, t_2 \in r \,/\, F(t_1[X], t_2[X]) \geq \alpha \wedge T(t_1[Y], t_2[Y]) \geq \beta\}}{Card\{(t_1, t_2)\ t_1, t_2 \in r \,/\, F(t_1[X], t_2[X]) \geq \alpha\}} \text{ Otherwise}$$

Where ∧ is the logical operator and. The basic idea consists of computing the percentage of tuples which fulfill the antecedent and consequent together with respect to those which only fulfill the consequent.

Definition 2. The relation R verifies an α–ß FFD X→FTY with support s, where s ∈ [0, 1], is defined as:

$$s = 0 \text{ if } n = 0$$

$$s = \frac{Card\{(t_1, t_2)\ t_1, t_2 \in r \,/\, F(t_1[X], t_2[X]) \geq \alpha \wedge T(t_1[Y], t_2[Y]) \geq \beta\}}{n} \text{ otherwise}$$

where *n* is the number of tuples of the *r* instance of the relation R.

The idea is to find the percentage of tuples which fulfill the antecedent and consequent together with respect to the total rows of the relation.

Another way of considering the connections between data in databases is to specify a relationship between objects in a dataset and reflect monotonicity in the data by means of that we have called gradual fuzzy dependencies (GRFDs). It is closely related to the idea of gradual rules introduced by Dubois and Prade [3]. An intuitive example of a GRFD is "the bigger business is the higher earnings they have" and we assume that the concept of GRFD can be considered, in this way, as similar to the FFD one. Therefore we define:

Definition 3 (α–ß gradual functional dependency). The relation *R* verifies an α–ß GRFD $X \int_{FT} Y$ if and only if, for every instance *r* of *R* it is verified:

$$\forall t_1, t_2 \in r,\ F'(t_1[X], t_2[X]) \geq \alpha \Rightarrow T'(t_1[Y], t_2[Y]) \geq ß$$

where F' and T' are fuzzy relations of the type: *fuzzy greater than, fuzzy greater than or equal to, fuzzy less than, fuzzy less than or equal to, fuzzy not equal,* etc. We can define an α–ß GRFD $X \int_{F'T'} Y$ with confidence *c* in the same way that we have made it for FFD (see Definition 1).

4 Applying FSQL to obtain fuzzy global dependencies

Now, it is necessary to relate the FSQL environment to our definitions. To do so, we first introduce a general definition of Fuzzy Global Dependencies based on FSQL operators and FSQL CDEG function, later we will show how FGD can be calculated with FSQL.

4.1 Fuzzy Global Dependencies with FSQL operators

Definition 4. The relation *R* with attribute sets $X=(x_1\ldots x_n)$, and $Y=(y_1\ldots y_m)$ whose attributes are trapezoidal possibility distributions, verifies an α–ß FGD $X \blacktriangleright_{F^*T^*} Y$ with $\alpha=(\alpha_1,\alpha_2,\ldots,\alpha_n)$ / $\alpha_i \in [0,1]$ $\forall i=1,\ldots,n$ and $ß=(ß_1,ß_2,\ldots,ß_m)$ / $ß_j \in [0,1]$ $\forall j=1,\ldots,m$, if and only if, for every instance *r* of *R* it is verified:

$$\forall t_1, t_2 \in r,\ \bigwedge_{i=1,2,\ldots,n}[F^*_i(t_1[x_i], t_2[x_i]) \geq \alpha_i] \Rightarrow \bigwedge_{j=1,2,\ldots,m}[T^*_j(t_1[y_j], t_2[y_j]) \geq ß_j] \text{ where}$$

F^*_i:UxU→[0,1]/F^*_i(A,B)=CDEG(A *F_Comp_ant$_i$* B)

T^*_j:UxU→[0,1]/T^*_j(A,B)=CDEG(A *F_Comp_con$_j$* B)

∀A=\$[$\alpha_A,\beta_A,\gamma_A,\delta_A$], B=\$[$\alpha_B,\beta_B,\gamma_B,\delta_B$] ∈ U (see Figure 1)

F_Comp_ant$_i$, F_Comp_con$_j$ defined as any fuzzy comparator in FSQL (any *F_Comp* in Table 1, even when preceded by a NOT operator) ∀i=1,…,n, ∀j=1,…,m

Definition 5. The relation *R* with attribute sets X=(x_1…x_n), and Y=(y_1…y_m) whose attributes are trapezoidal possibility distributions, verifies an α–ß FGD X$\blacktriangleright_{F^*T^*}$Y with α∈[0,1] and ß ∈[0,1], if and only if, for every instance *r* of *R* it is verified:

$$\forall t_1, t_2 \in r,\ \wedge_{i=1,2,\ldots,n}[F^*_i(t_1[x_i], t_2[x_i]) \geq \alpha] \Rightarrow \wedge_{j=1,2,\ldots,m}[T^*_j(t_1[y_j], t_2[y_j]) \geq ß]$$

$$\forall i=1,\ldots,n,\ \forall j=1,\ldots,m$$

Now, we can make a new definition of FFDs and GRFDs as a particular case of FGDs.

Definition 6. If *F_Comp_ant$_i$, F_Comp_con$_j$* ∈ {FEQ,NFEQ} then we say that R verifies an α_i–$ß_i$ FFD X$\rightarrow_{F^*T^*}$Y.

Definition 7. If *F_Comp_ant, F_Comp_con* are any *F_Comp* of FSQL such that there exists at least a *k* from 1 to *n* which fulfils *F_Comp_ant$_k$* ∉ {FEQ,NFEQ} and at least a *s* from 1 to *m* which fulfils *F_Comp_con$_s$* ∉ {FEQ,NFEQ} then we say that R verifies an α–ß GRFD X$\int_{F^*T^*}$Y.

Of course we can define an α–ß FGD X$\blacktriangleright_{F^*T^*}$Y with confidence *c* in the same sense that we have made it for FFD (see Definition 1). To simplify notation, in X$\blacktriangleright_{F^*T^*}$Y we will denote F^* as *(F_Comp_ant$_i$)** ∀i=1,…,n, and similar notation for T*.

4.2 Obtaining Fuzzy Global Dependencies using FSQL

Let *R* be a relation with attribute sets X=(x_1…x_n), Y=(y_1…y_m) and PK=(pk_1…pk_S) included in its scheme, where PK is the primary key of *R*. To determine if *R* verifies an α–ß FGD X$\blacktriangleright_{F^*T^*}$Y for an instance *r*, we create a FSQL query with the following general format:

```
SELECT count(*) FROM   r A1, r A2
WHERE  (A1.PK <> A2.PK)
AND        (A1.x1 F_Comp_ant1 A2.x1 THOLD α1
                          …AND…
                 AND A1.xn F_Comp_antn A2.xn THOLD αn)
AND NOT    (A1.y1 F_Comp_con1 A2.y1 THOLD ß1
                          …AND…
                 AND A1.ym F_Comp_conm A2.ym THOLD ßm)
```

The basic idea consists of computing the tuples which fulfill the antecedent and do not fulfill the consequent. Therefore, if the result of the query is 0, we can say that *R* verifies FGD for the instance *r*.

If the result of previous counting is not 0, we can determine if *R* verifies an α–ß FGD X$\blacktriangleright_{F^*T^*}$Y with confidence *c* by means of a simple procedure as follows (algorithm 1):

Step 1.1: To obtain the value *a* as the number of tuples which fulfil the antecedent and consequent together:

```
SELECT count(*) FROM   r A1, r A2
WHERE  (A1.PK <> A2.PK)
 AND       (A1.x1 F_Comp_ant1 A2.x1 THOLD α1
                          …AND…
                 AND A1.xn F_Comp_antn A2.xn THOLD αn)
 AND       (A1.y1 F_Comp_con1 A2.y1 THOLD ß1
                          …AND…
               AND A1.ym F_Comp_conm A2.ym THOLD ßm)
```

Step 1.2: To obtain the value *b* as the number of tuples which fulfil the antecedent:

```
SELECT count(*) FROM   r A1, r A2
WHERE  (A1.PK <> A2.PK)
 AND       (A1.x1 F_Comp_ant1 A2.x1 THOLD α1
                          …AND…
                 AND A1.xn F_Comp_antn A2.xn THOLD αn)
```

Step 2: To obtain the degree of confidence *c* as *c=a/b*.

Step 3: To determine if the computed degree indicates that the FGD is good enough, we can compare the value *c* with some fuzzy quantifier defined in the FMB (by example *most*).

Notice that FSQL also allows us to compare (with fuzzy comparators) crisp attributes. In order to do this, FSQL makes a fuzzyfication of the crisp value before the comparison, transforming it into a triangular possibility distribution (according to values stored in the FMB for the attribute). This fuzzyfication can either be implicit or explicit (with the fuzzy constant #). Also, FSQL can work with scalar attributes but with them we can use only use the comparator FEQ (because an order relationship in their domains is not defined).

If the purpose is to search for FFDs in order to discover intentional properties (constraints that exist in every possible manifestation of the database frame) it seems more appropriate to use a weak resemblance measure in the antecedent (FEQ, based on possibility) as a fuzzy comparator and a strong one in the consequent (NFEQ, based on necessity). In this way, we get interesting properties which can help us with the decomposition of relations. Searching for FFDs or GRFDs to discover extensional properties (those existing in the current manifestation of the data) is a task for DM. In this case, the choice of the fuzzy comparators and the parameters α, ß we will be made according to the specific problem in question.

5 Finding preferences using FSQL

In this section we are going to apply the process detailed in the previous section to estimate the significance that temperature evolution through a day has in the decision-making process for selecting the best day to soar.

The process shown here is part of the DSS system outlined in [1]. That work was centered in the design of a DSS for soaring site recommendation, developed in order to be offered as an added-value service by travel agencies. Site selection process for soaring depends on a number of factors, resulting in a complex decision-making task. It is common for the decision makers to use their subjective judgment and previous experience when selecting the most appropriate place for soaring or gliding (soaring is the correct term to use when the craft gains altitude or speed from movements of the atmosphere during the flight). The reason is that data for place selection originate from varied sources and are not organized in a format that decision makers can acquire any meaningful information. To solve this problem, the integration of a DW and a DSS seems to be efficient to help retrieve data from different databases and information sources and analyze them in order to provide useful and explicit information.

Table 1: TEMP_VS_QUALITY table

Day	Min_temp	Max_temp	Avg_temp	Score
1	28,7	37,6	32,7	1161
2	27,1	39,6	34,51	1161
3	6,8	8,55	7,75	3317
4	29,6	38,07	31,3	2453
5	8,02	9,8	9,21	3151

Let suppose a tourism manager who wants to know if the values of the temperature evolution, in terms of maximum, minimum and average value, has been taking into account by the experts to decide whether or not to soar: let TEMP_VS_QUALITY be a relation with the minimum, average and maximum temperature, and the score given by the experts about the quality of the days with the data shown in Table 1. This table has been obtained from a Data Warehouse system which integrates the historical weather information of different sites for soaring. In this case, the table corresponds to data obtained from province of Granada. Our objective is to determine “if similar behavior with respect to temperature (minimum, average and maximum) implies similar day quality”. To manage these attributes we use:

- Minimum/maximum temperature: is the minimum/maximum value of temperature registered in a day. This are crisp attributes but we decide define them as Type 1 in the FMB using the fuzzy constants value #n, which means “approximately n” (represented by triangular possibility distributions). These values correspond to tags (1) and (2) in figure 1.

- Average temperature: is the average temperature value of temperature registered in a day. Although it is a crisp attribute we decide to define it as Type 1 in the FMB as well as minimum and maximum attribute detailed previously. This value corresponds with tag (3) in figure 1.

- Score: this is the quality value of the day given by the experts. The final value is calculated analyzing the weather conditions from different points of view (temperature, wind, pressure…). As well as previous attributes, we have decided to define it as Type 1 in the FMB.

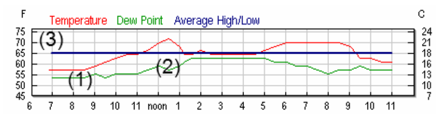

Figure 1: Temperature analysis.

In the FMB we have defined margin=6 for *Min_temp*, *Avg_temp* and *Max_temp* and margin=400 for *Score*. After some trials we show the results obtained (Step 1.1 and 1.2 of Alg. 1) in Figure 2. Therefore (Step 2 of Alg. 1) we can say that TEMP_VS_QUALITY verifies:

(0.6,0.7,0.6)–(0.5) FFD (Min_temp, Avg_temp, Max_temp) $\rightarrow$ (FEQ)*(NFEQ)* (Avg_score)

with confidence c=2/3. If we compare this value with the fuzzy quantifier most (Step 2) we can say that the FFD is verified with fulfillment thresholds 0.56 for most of the tuples.

Now (Step 3 of procedure in 4.2) if we compare this value with the fuzzy quantifier *most* we can say that the above FGD is verified with fulfillment thresholds 0.78 for most of the tuples. We can conclude that the temperature values detailed previously in this point are taken in consideration by the experts in the overall decision-making process of soaring site selection, so it would be desirable that the DSS developed take into account this situation and develop tools to mange correctly this kind of information.

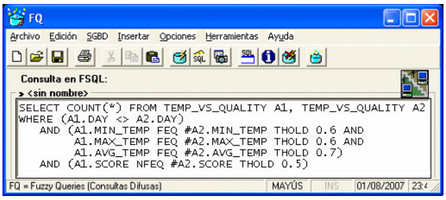

Figure 2: Result of FSQL query Step 1.1 Alg.1 is: 3.

6 Conclusions

In this paper we have shown how a Fuzzy Data Mining process can help tourism agencies in the design of their DSS. The process outlined in this paper helps to find out hidden and subjective preferences which experts use for decision-making complex tasks. Actually, we have detailed the process to estimate the significance that temperature evolution through a day has in the decision-making process for selecting the best day to soar.

This DM process is based on the use of Fuzzy Global Dependencies (FGDs) as a common framework to integrate fuzzy functional dependencies and gradual functional dependencies. Also, we have relaxed the FGD definition for finding FGDs even if exceptional tuples do not verify it. FGDs are defined with the FSQL fuzzy comparators on trapezoidal possibility distribution. Therefore, the FSQL language is the natural way to obtain such FGDs. Using possibility in FGDs as a weak resemblance in the antecedent and necessity as a strong one in the consequent, FSQL could be used to find FFDs which portray constraints that exist in every possible manifestation of the frames in a database (useful for the decomposition of relations). A practical application is to search for FGDs in order to discover properties which exist in the current manifestation of the data as a task for DM.

Acknowledgements

This work has been supported by the Spanish Research Program under projects TIN2006-02121 and TIN2005-09098-C05-03 and by the Andalucía Research Program under projects JA031/06 and GR050/06.

References

[1] Araque, F., Salguero, A., Abad, M.M.: Application of data warehouse and Decision Support System in Soaring site recommendation. *Proc. Information and Communication Technologies in Tourism*, ENTER 2006. Springer Verlag, Lausanne, Switzerland.

[2] Carrasco, R., Vila, M. A., Araque, F. "dmFSQL: a Language for Data Mining," dexa, pp. 440-444, 17th *International Conference on Database and Expert Systems Applications* (DEXA'06).

[3] D. Dubois, H. Prade, "Gradual rules in approximate reasoning". Information Sciences, Vol 61, pp. 103-122.

[4] Ghotb, F., & Warren, L. (1995). A case study comparison of the analytic hierarchy process and a fuzzy decision methodology. *Engineering Economist*, 40(3), 233–246.

[5] Gretzel U., Mitsche, N., Hwang, Y.-H., Fesenmaier, D. (2004). "Tell me who you are and I will tell you where to go – Use of travel personalities in destination recommendation systems." *Information Technologies and Tourism* (IT&T), Vol 7.

[6] H. Mannila, K. Räihä, Dependency inference, in: *Proc. 13th VLDB Conf.*, 1987, pp. 155–158.

[7] J. Galindo, J.M. Medina, O. Pons, J.C. Cubero, "A Server for Fuzzy SQL Queries", in "*Flexible Query Answering Systems*", eds. T. Andreasen, H. Christiansen and H.L. Larsen, Lecture Notes in Artificial Intelligence (LNAI) 1495, pp. 164-174. Ed. Springer, 1998.

[8] J.C. Cubero, M.A. Vila, "A new definition of fuzzy functional dependency in fuzzy relational databases" *International J. Intelligent Systems*, Vol 9, pp. 441-449, 1994.

[9] Lexhagen, M. (2005). The importance of value-added services to support the customer search and purchase process on travel websites. Information Tech. and Tourism, 7(2), 119-135.

[10] Liang, G., & Wang, M. (1991). A fuzzy multi-criteria decision making method for facility site selection. International *Journal of Production Research*, 29(2), 2313–2330.

[11] T. Akutsu, A. Takasu, Inferring approximate functional dependencies from example data, in: Proc. AAAI93 *Workshop on Knowledge Discovery in Databases*, AAAI, New York, 1993, pp. 138–152.

[12] Xiang, Z. and D. R. Fesenmaier (2005.) An analysis of two search engine interface metaphors for trip planning, *Journal of Information Technology and Tourism*, 7.