Knowledge-Based Systems 290 (2024) 111574

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

An extended proximity relation and quantified aggregation for designing robust fuzzy query engine

Miroslav Hudec [a,b,*], Miljan Vučetić [c,d], Nina Barčáková [b]

[a] Faculty of Economics, VSB – Technical University of Ostrava, Ostrava, Czech Republic
[b] Faculty of Economic Informatics, University of Economics in Bratislava, Bratislava, Slovakia
[c] Vlatacom Institute of High Technology, Artificial Intelligence Department, Belgrade, Serbia
[d] Faculty of Informatics and Computing, Singidunum University, Belgrade, Serbia

ARTICLE INFO

Keywords:
Proximity relation properties
Conformance measure
Fuzzy number
Fuzzy quantifier
Aggregation function
Suggesting products

ABSTRACT

In this article, we propose a novel model of a robust fuzzy query engine that addresses vagueness in data and users' requirements. It aims to assist users in recommending similar products or services by retrieving the most suitable entities when the limitations of queries and recommendation approaches are recognized. The proposed fuzzy engine model considers various complex aspects, including imprecise preferences explained by linguistic terms, uncertain data in datasets, connections among elementary requirements, and the lack of historical data necessary for personalized recommendations. To achieve this goal, we propose a similarity matching based on the extended proximity relation and an adapted conformance measure for elementary requirements. In this direction, a new monotonicity property for proximity relation is introduced to ensure consistency in similarities among ordinal categorical data (including binary data) expressed by crisp or fuzzy numbers. Therefore, the conformance measure used to evaluate the similarity between user requirements and attribute values is expressed as a fuzzy number. Next, we propose a quantified aggregation of elementary requirements by strictly monotone fuzzy relative quantifiers. The flexibility is further extended by a convex combination of possibility and necessity measures. A hotel selection experiment is being carried out to explore the potential of the proposed fuzzy query engine. Finally, the limitations and usefulness of the proposed approach are addressed.

1. Introduction

With a present-day plethora of similar services or products, and attributes explaining them, it has become a demanding task to find or recommend the most suitable services or products [1,2]. Data explaining products or services usually are in databases that represent so-called *hard* information, i.e., crisp or precise data. For instance, the number of days with snow coverage, an average monthly temperature in Celsius degrees, binary data (e.g., whether a hotel room has a balcony), categorical values from the Likert scale expressing an opinion, and so on. It is obvious that some of the data are precise, whereas others are imprecise by nature [3], but stored as crisp values [4].

A relational database can be formalized as a set of pairs as follows [5]:

$$T_e = (U_e, A_e) \tag{1}$$

where U_e is an universe of records (entities), A_e is a set of attributes in a relation T_e, $e = 1, \dots, n$. In this relation, rows express records, while columns represent attributes.

On the other side are customers searching for the most suitable entities, for instance, an accommodation unit. They prefer explaining preferences by linguistic terms, like *distance is rather short*, *altitude above sea level around 900 m*, *a slight preference for a balcony*, or aggregated requirements like *the majority of posed conditions should be met, when the price is acceptable*. These requirements represent so-called *soft* information. An example is when people add requirements like the sustainability support and pollution. Usually, these requirements are not mandatory, but influence ranking. The next example is financial aspect. When the price is slightly beyond the sharp border, but the other requirements are very nicely met, should we reject such an alternative, or rank it as an acceptable, but not the ideal one?

Traditional Boolean logic allows users to formulate precise queries using exact matches and logic aggregation operators restricted to the values 0 and 1. The literature has already advocated the limitations of query and recommendation approaches based on two-valued logic and provided solutions for diverse situations. The limitations include

* Corresponding author at: Faculty of Economic Informatics, University of Economics in Bratislava, Bratislava, Slovakia.
E-mail addresses: miroslav.hudec@vsb.cz (M. Hudec), miljan.vucetic@vlatacom.com (M. Vučetić), nina.barcakova@euba.sk (N. Barčáková).

https://doi.org/10.1016/j.knosys.2024.111574
Received 25 August 2023; Received in revised form 5 January 2024; Accepted 24 February 2024
Available online 28 February 2024
0950-7051/

their sensitivity to exact matches, which may not be suitable for scenarios where data or preferences are imprecise, uncertain, or contain variations. Fuzzy queries introduce a level of flexibility and tolerance for imprecision. They allow selecting records based on their matching degrees rather than requiring an exact match. The foundation for flexible approaches covering requirements expressed vaguely is the theory of fuzzy sets introduced in [6], the fuzzy logic theory based on the concepts of fuzzy sets and many-valued logics [7], the proximity relation and conformance function devoted to mining patterns in data and recognizing redundancies [2,8,9] and the theory of logic aggregation functions explored e.g., in [10–13].

Data might be also stored in fuzzy databases or in a less restrictive structure like an Excel file, where attributes can assign a subset of categorical values. The literature has already recognized this situation and therefore various solutions were proposed (see, e.g., [2]), including recommending less-frequently bought items, when recommendation systems face a low amount of historical data to provide a reliable recommendation.

However, in this area, the problems of the complete consistency in proximity relation, [14], soft requirements over binary data [15], limitations of conjunctive or disjunctive aggregation, and adjustment to the properties of human reasoning [11] remain.

In our work, we aim to address research gaps identified in existing studies, particularly those related to flexibility in matching among categorical data (including binary data), consistency in expressing their similarities, adaptability to mixed data types, a lack of proximity expressed linguistically, adapting conformance to fuzzy data, and their aggregation in a holistic manner.

The main contributions of our work are:

1. proposed mathematical property of monotonicity for obtaining the consistent matrix of proximity measures among ordinal categorical data, keeping the usual properties of human reasoning,
2. for proximities expressed by linguistic terms represented through triangular fuzzy numbers, the Ruspini partition is introduced to support the proposed mathematical property of monotonicity,
3. proposed aggregation of elementary conformances based on these proximities by quantified aggregation extended by a convex combination of possibility and necessity measures to cover the whole scale of users from low-demanded ones to high-end ones.

Fuzzy query engines are important because they enable more flexible and tolerant selection or retrieval mechanisms, especially when dealing with imprecise data or approximate requirements. They find applications in tasks such as recommendation systems, information retrieval, ranking, and scoring. This flexibility generally helps users identify the most relevant results when an exact match is not found or when users do not have precise information about what they are looking for. Our contributions are exemplified through theoretical advancements and demonstrated in a hotel selection experiment detailed in this paper.

The remainder of this article is organized in the following way. Section 2 introduces theoretical concepts and related works. Section 3 is devoted to the proposed approach, whereas Section 4 is focused on the experiment. Section 5 discusses the proposed approach and raises future research to overcome the recognized limitations. Finally, Section 6 conveys the concluding messages.

2. Related works

This section is devoted to the preliminaries in the fields of proximity and similarity relations, aggregation functions, and possibility and necessity measures which represent the starting point of this research. Before we move to these fields, a convenient and cogent introduction is provided.

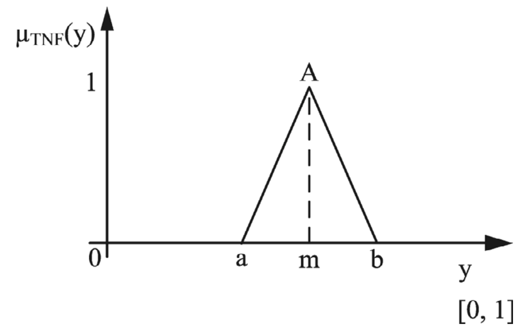

Fig. 1. Graphical interpretation of triangular fuzzy number $A(a, m, b)$.

A fuzzy set A is a subset of the universe X formalized by the membership function $\mu_A(x)$, which computes the membership degree $\mu \in [0,1]$ for $\forall x \in X$ [6].

Two types of fuzzy sets expressed by piecewise linear functions are applied in this work: the triangular fuzzy number (TFN) shown in Fig. 1 for handling proximity and conformance, and the linear gamma function for formalizing fuzzy relative quantifiers (see, Fig. 3). Next, the operations of addition between two fuzzy numbers and multiplication by a parameter are introduced. Finally, possibility and necessity measures as well as dominance relation are introduced.

Two different approaches are used for fuzzy arithmetic on fuzzy numbers: (i) the α-cuts and interval calculation approach (so-called standard fuzzy arithmetic), and (ii) the Zadeh extension principle approach using t-norms. The former might lead to an overestimation of uncertainties in the resulting fuzzy numbers [16,17]. This is why we adapted the latter. When two fuzzy numbers A and B are expressed by the continuous membership functions $\mu_A(x)$ and $\mu_B(y)$, respectively, the Zadeh extension principle gives the membership function for the fuzzy set $C = A \circledast B$ as [18,19]

$$\mu_C(z) = \mu_A(x) \circledast \mu_B(y) = \sup_{z=x\star y} \min(\mu_A(x), \mu_B(y)) \tag{2}$$

where $\circledast$ stands for any of the fuzzy arithmetic operations, and $\star$ stands for any of the related arithmetic operations implemented on crisp numbers. As a simple example, consider the addition of two fuzzy singletons 3 and 5. The matching degree of the value 4 to both singletons is zero, then the degree is zero for 4 + 4, but for 3 + 5, it is 1. For all other additions, the matching degree is 0. Thus, the solution is a fuzzy singleton 8. TFNs are closed under the extended addition using the minimum t-norm, i.e., the result is a triangular fuzzy number.

If TFNs A and B are expressed by the triplets (a_A, m_A, b_A) and (a_B, m_B, b_B), respectively (see the notation in Fig. 1), then the fuzzy addition $A \oplus B$ is simplified as:

$$A \oplus B = (a_A + a_B, m_A + m_B, b_A + b_B) \tag{3}$$

while the multiplication with parameter $r \in \mathbb{R}$ is realized as:

$$r \cdot A = (r \cdot a_A, r \cdot m_A, r \cdot b_B) \tag{4}$$

The possibility measure (*Pos*) expresses to what extent it is possible that a fuzzy set (in our case a TFN A) belongs to the fuzzy concept (in this case quantifier Q) as [20]:

$$Pos(A, Q) = \sup_{x\in X} \min(\mu_A(x), \mu_Q(x)) \tag{5}$$

The necessity measure (*Nec*) expresses to what extent it is necessary that a TFN A belongs to the fuzzy concept Q as [20]:

$$Nec(A, Q) = \inf_{x\in X} [\max(\mu_Q(x), 1 - \mu_A(x))] \tag{6}$$

Observe that the possibility measure is optimistic in nature, whereas the necessity one is a pessimistic view. Moreover, the necessity measure

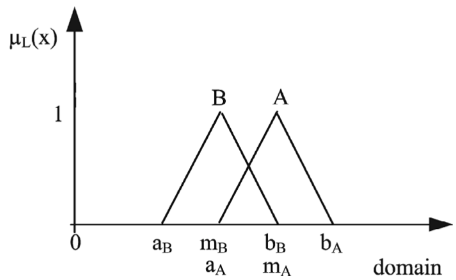

Fig. 2. Fuzzy number A dominates fuzzy number B by (7).

is significantly restrictive, which might lead, in many cases, to degrees equal 0 and therefore to the clear rejection of the evaluated entities.

Next, we need to compare fuzzy numbers to find the minimal or maximal one for the conformance measure explained later on. More details about comparing fuzzy numbers are in, e.g., [21]. A fuzzy set A dominates fuzzy set B when:

$$dominance(A,B) = \frac{P(A \succ B) + P(A \sim B)}{P(A) + P(B)} > 0.5 \tag{7}$$

where $P(A \succ B)$ is an area where A dominates over B, $P(A \sim B)$ is an area where they are indifferent, and $P(A)$ and $P(B)$ are areas of fuzzy sets A and B, respectively. Fuzzy set A dominates B when $dominance(A,B) > 0.5$, see Fig. 2.

Finally, we need a linguistic variable (LV), which takes terms instead of numbers. A LV is defined by a quintuple (L, $T(L)$, X, G, H), where [22]:

- L stands for the variable name,
- $T(L)$ is a set of linguistic labels related to the variable L,
- X is the universe of discourse,
- G is the syntactic rule to generate $T(L)$,
- H is the semantic rule that assigns each label in $T(L)$ to its meaning $H(L)$.

In order to have linguistic variables of a suitable quality for our work, it should be a fuzzy partition [23] and meet the quality criteria raised in [24].

Fuzzy query approaches are mainly focused on offering flexibility to users when they are unable to express clear requirements for elementary conditions and their aggregation. Generally, for the calculation of matching degrees of the elementary conditions and the aggregation of the elementary conditions are two main parts. The first part generally adopts fuzzy sets for concepts and further enhances flexibility by possible weights of elementary conditions. For conjunctive queries, the weights should be managed by fuzzy implications [25,26], because all elementary conditions are mandatory. The second part usually covers the symmetric conjunctive and disjunctive aggregation as well as their limitations [27]. Asymmetry is handled by, e.g., non-commutative aggregation queries like constraints and wishes (Bosc and Pivert, 2012; Hudec and Mesiar 2020) or the so-called *among* aggregation. The next topic is focused on user-friendly interfaces [28] to cover diverse needs. Finally, empty or over-abundant answer problems might appear and therefore they should be managed [29,30]. Thus,conformance and proximity with the suitable aggregation might bring further benefits.

In the next two subsections, we introduce proximity relation, conformance measure, and quantified aggregation.

2.1. Proximity relation and conformance measure

Proximity relations are fundamental in detecting redundancies inside fuzzy databases and in various data analysis tasks where assessing data similarity is crucial. The idea of extending the queries with a conformance measure based on the proximity relation was proposed in [2]. Proximity relation is a binary relation indicating the degree to which two values are approximately equivalent. It provides a general numerical representation of similarity. Thus, the properties of proximity relation are general [8] to cover similarity between categorical data regardless they are nominal or ordinal. Imagine that the user expresses requirement by terms *low* or *medium*, while in a data set the record value for this attribute is numerical in the borderline case of categories *medium* and *high*. This is neither a clear match nor a clear rejection. Thus, the proximity relation is relevant and therefore should be examined and improved to solve the requirements for keeping consistency among ordinal categorical data effectively.

Recent literature also addresses the need for relaxed functional dependencies [31]. It is beneficial when the data and users' requirements are imprecise. It explores natural and different similarities between categorical and numerical data [14] and acknowledges the existence of similarity between binary data when expressing elementary requirements linguistically. We should also consider scenarios involving more values for a single attribute when data are not in a relational database (e.g., in a spreadsheet) or are in a fuzzy database. In this case, the usual proximity measures cannot be applied, or we need several different similarity functions for one elementary requirement. Recently, this relation becomes one of the key functions for mining useful patterns by fuzzy functional dependencies from crisp or fuzzy databases [2,14,31].

Definition 1. The proximity relation is defined as a mapping s : $D \times D \to [0,1]$ where for each $x, y \in D$ holds [8]:

- $s(x,x) = 1$ (i.e., reflexivity)
- $s(x,y) = s(y,x)$ (i.e., symmetry)

where D is a domain of the attribute, e.g., a set of categorical values.

Next, let s be a proximity relation on domain D, then for any $\alpha \in (0,1]$, the elements $x, y \in D$ are α similar if and only if $s(x,y) \geq \alpha$ [8].

For instance, when a user poses requirement *A is very low*, then an entity having the value of attribute A expressed as *low*, is not an ideal match (i.e., $0 < s(\text{low}, \text{very low}) < 1$), but clearly a better option than an entity having value *medium*. This concept also supports conformances when a user's requirement is expressed as a subset of categories (e.g., {*very low*, *low*}), while a database stores one categorical value to keep the requirement of normality in classical relational databases (see e.g., [32]), or even when a fuzzy database stores two neighbouring categories [33].

The proximity relation is the extension of the similarity relation proposed by Buckley and Petry [34], when the max–min transitivity is excluded. The max–min transitivity is a desired property in clustering, but it does not fully meet the requirements for proximity in conformance measure [9] and in fuzzy queries [15]. The similarity relation has been proposed for fuzzy relational databases to handle identity relations, i.e., to recognize and eliminate redundancies [8]. Let us suppose that A is similar to B to a level of 0.75, and B is similar to C to a level of 0.75. Then, by the max–min transitivity: $s(x,z) \geq \max_{y \in D} \min[s(x,y), s(y,z)]$, the similarity between A and C cannot be less than 0.75. It is a suitable way for clustering similar data in a fuzzy database, but not for queries. Let *dark brown* be similar to *brown* with 0.8 and *brown* with *light brown* with 0.8, then similarity between *dark brown* and *light brown* should be less than 0.8 (i.e, $s(\text{dark brown}, \text{brown}) > s(\text{dark brown}, \text{light brown})$).

The proximity relation introduced in [8] (relaxed to avoid max–min transitivity) is suitable for expressing numerical similarities among nominal categorical data, because they do not have a natural partial ordering. However, for ordinal categorical data, the situation differs.

Usually, users directly assign proximity as real numbers. These values might be also learned by machine learning approaches. Generally, it works when people assign values, but it is not straightforward for machines. It means that we should introduce further properties to keep consistency. The monotonicity of the proximity values between categories should reflect their partial order, i.e., the proximity between *dark brown* and *brown* should be higher or equal than the proximity between *dark brown* and *light brown*.

Numerical proximities provide space for precisely expressing similarities between categorical data. For many tasks, it is a desired support. However, in many other tasks, people cannot precisely express it, e.g., is it 0.72 or 0.78? Therefore, a consistent family of linguistic terms explaining proximity should be created.

The next example is the binary data in a database and users inclination towards these extremes, e.g., *a slight preference of balcony*, where the database contains values from the domain $D = \{Yes, No\}$. This observation is related to the Law of Excluded Middle which says that every proposition could only be true or false, i.e., nothing in between (in our example, question: Do you want a balcony or not). An earlier version of this law, proposed by Parmenides in approx. 400 BC, met with quick reaction. Heraclitus, a fellow philosopher, declared that logical propositions could simultaneously be true and false [35]. The proximity relation having a set of suitable properties is a way to formalize it.

The next required concept is the conformance measure. The equation for expressing the conformance between the categorical record in a database r_i and a user's preferred value r_j is as follows [9]:

$$C(X_k[r_i, r_j]) = \min\left\{ \min_{x \in Di}\left\{ \max_{y \in Dj}\{s(x,y)\}\right\}, \min_{x \in Dj}\left\{ \max_{y \in Di}\{s(x,y)\}\right\}\right\} \tag{8}$$

where D_i is the list of values (domain) of attribute X_k for record r_i, D_j is the list of values (domain) of attribute X_k for the user requirement r_j, $D_i, D_j \in D$, $s(x, y)$ is a proximity relation for the values x and y in domain D (see Definition 1). This equation was initially proposed for computing the conformance between each two records in a database to recognize strong redundancies or reveal relational dependencies.

This equation is convenient for categorical data where attributes' values are described by linguistic terms on discrete domains. This equation is also suitable for the numeric data after their discretization [36] or fuzzification [15].

2.2. Quantified aggregation

An aggregation function combines the values of input requirements into a single output value, considering all inputs. The final ranking of alternatives or records can be realized by the Simple Additive Weighting method [21], which is a weighted arithmetic mean. The next option is conjunctive aggregation, where all atomic requirements should be at least partially satisfied. These two relatively simple methods can be used for aggregating conformances, but they are not suitable for all cases [2,11].

The standard classification of aggregation functions is as follows [37]: conjunctive functions are characterized by $Af(\mathbf{x}) \leq \min(\mathbf{x})$, disjunctive functions by $Af(\mathbf{x}) \geq \max(\mathbf{x})$, averaging functions by $\min(\mathbf{x}) \leq Af(\mathbf{x}) \leq \max(\mathbf{x})$, and the remaining functions are mixed, where $\mathbf{x}$ is a vector, i.e., $\mathbf{x} = (x_1, \dots, x_n)$.

Next, aggregation functions meets boundary conditions:

$$Af(0, 0, \dots, 0) = 0 \text{ and } Af(1, 1, \dots, 1) = 1 \tag{9}$$

and monotonicity

$$y \leq z \Rightarrow Af(x, y) \leq Af(x, z) \tag{10}$$

When all elementary requirements should be satisfied, the solution is a conjunctive function. However, all conjunctive functions, except the minimum t-norm have downward reinforcement property [38]. This means that the solution for a larger number of elementary conditions (such as evaluating a hotel) is significantly lower than the lowest one. (i.e., when ten elementary conditions are satisfied with a degree of 0.5 each, the product t-norm computes an output approx. 0.00098, while for Łukasiewicz t-norm the output is 0). On the other hand, the minimum t-norm does not consider higher values than the minimal one

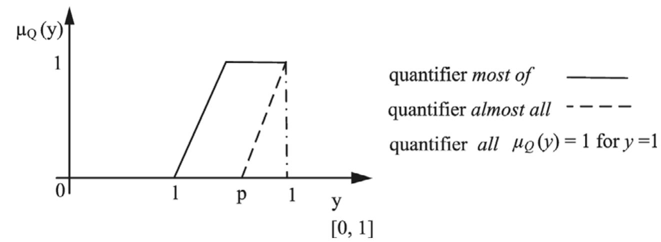

Fig. 3. Parametrized fuzzy relative quantifier *most of*, where y expresses the proportion of the satisfied predicates or elementary requirements P_i (see, Eq. (11)).

(that is, min(0.2, 0.21) = min(0.2, 0.8)). In disjunctive aggregation, when at least one condition is significantly met, the solution is close to or equal to 1 (the dual upward reinforcement property), which is a very relaxing aggregation, i.e., in queries it might lead to the overabundant answer problem. Thus, the solution for this problem seems to be an averaging function or a mixed behaviour function. Regarding the former, a conjunctively polarized aggregation function with an annihilator of 0 [11] could be the solution. However, when an entity (e.g., a hotel room) meets all elementary requirements with a very low degree, say 0.15, we tend to reject such a solution. Due to the internality property of the averaging functions [39], the solution is not 0. Hence, the solution seems to be a mixed aggregation function. Thus, the uni-norm appears as an immediate solution. Although, associativity is met, it is not the best option due to the non-continuity in points {0, 1} and {1, 0} [10,40]. Finally, the recently proposed ordinal sums of conjunctive and disjunctive functions [41], might be the solution, because of their nature to express an inclination to the clear acceptance. However, they are quite complex for end users and domain experts (e.g., travel agents and customers) to recommend products due to a higher number of parameters to assign. Thus, the option can be quantified aggregation which covers the evaluation: "to more to merrier" [12].

In this section, we introduce quantified aggregation, which should be adjusted to the user linguistically interpreted elementary conformances. In this approach, satisfying a conformance is optional, but as the number of satisfied conformances (and their intensities) increases, the recommendation increases as well. Such aggregation is of the structure *the most of elementary requirements $\{P_1 \dots P_n\}$ should be met* [42]. It is formalized as:

$$v(x) = \mu_Q\left(\frac{1}{n}\sum_{i=1}^{n} \mu_{P_i}(x)\right) \tag{11}$$

where n is the number of elementary requirements, $y = \frac{1}{n}\sum_{i=1}^{n} \mu_{P_i}(x)$ is the proportion of elementary requirements P_i that are satisfied by the entity x, and μ_Q formalizes the quantifier *most of*. The truth value v gets values from the unit interval.

Within the sigma–count method, the fuzzy relative quantifier *most of* is formalized by increasing function [43]. When formalized by parameters, this quantifier yields [12] (see Fig. 3):

$$\mu_Q(y) = \begin{cases} 1, & \text{for } y \geq p \\ \frac{y-l}{p-l}, & \text{for } l < y < p \\ 0, & \text{for } y \leq l \end{cases} \tag{12}$$

where $0.5 \leq l \leq p \leq 1$. For $l = p = 1$ we get the crisp quantifier *all*, whereas for $0.8 \leq l < p = 1$ the function formalizes the fuzzy relative quantifier *almost all*.

This is an aggregation function, because it meets the axioms of boundary conditions and monotonicity. In addition, this quantified aggregation behaves like a mixed aggregation function (see, e.g., [12]). Behaviour depends also on the assigned parameters to quantifier *most of* depicted in Fig. 3 [40].

When working with fuzzy relative quantifiers in aggregation, we should evaluate whether monotonic functions for the quantifier *most*

Table 1
Proximity relation between binary values *Yes* and *No*, where a is an element in the unit interval indicating the vagueness in preference.

s(Yes, No)	Yes	No
Yes	1	a
No	a	1

of (and variants thereof) proposed in [25,43] and discussed in [44] are suitable, or they should be strictly monotone. The aggregation of fuzzy numbers by fuzzy quantifiers can be computed by the possibility and necessity measures [20]. Thus, there is a need for their convex combination to cover further properties in human evaluation: from high-end to low-demand customers.

3. The proposed approach

A common task in database retrieval and evaluation of entities for, e.g., recommending the most suitable ones is calculating the elementary requirements satisfaction and aggregating them in a way which meets the observable properties of human evaluation [11]. This section is devoted to developing a framework for evaluating and suggesting entities (products or services) without the need for personalization, storing users' behaviour and historical data.

3.1. Elementary requirements handled by the proximity relation

We adopt this function for two kinds of elementary requirements: linguistic preferences on binary data and proximity between categorical data. We explore how to express proximities by numbers and linguistic terms.

The boundary requirement is directly satisfied from the definition of the proximity relation (see Definition 1), i.e., $0 \le s(x,y) \le 1$.

This requirement keeps calculations in the expected range, e.g., for the α similarity [8]. It is also required when expressing proximity by linguistic terms and formalizing them by TFNs.

The proximity relation is adaptable and therefore is able to handle the query posed by a *soft* user condition on the binary attribute in a data set.

An example of a simple proximity relation is illustrated in Table 1, where $a \in [0,1]$. We recognize three main cases:

- user clearly wants one of the options, e.g., a hotel room must contain a balcony, yielding to $a = 0$,
- the user is absolutely indifferent, yielding to $a = 1$, or this elementary condition is excluded,
- the user has not a clear preference, yielding to $a \in (0,1)$.

The question is how to assign a value to the parameter a (Table 1) in the third case. Due to the symmetry, it holds $s(Yes, No) = s(No, Yes)$. Values closer to 0 or 1 emphasize almost clear opinions, while those close to 0.5 indicate a very vague requirements in the sense *I prefer a balcony, but without it, I will more or less accept room*. When the user strongly prefers a balcony, but will accept a room without one, then the penalization decreases the elementary requirement satisfaction to, e.g., $a = 0.25$. In the opposite case, when the user lightly prefers a balcony, the proximity is, e.g., $a = 0.75$, i.e., the penalization is light. In this example, we assign values of 0.25, 0.5, or 0.75 by the rule of thumb. More sophisticated ways can be applied, like learning from historical data. But it does not influence the axioms of proximity relation or further calculations.

The proximity relation proposed in [8] is relaxed to avoid max-transitivity (Section 2.1), but it does not fully meet the requirements for proximity in queries. For instance, when a user poses a query by term *very low*, it is clear that the proximity with term *low* should be higher than the proximity with term *medium*. To keep this approach consistent with the properties of human evaluation, we propose a new axiom for the proximity relation $s(x,y)$. It is the axiom of monotonicity for ordered ordinal categorical values like terms {very short, short, medium, large, quite large, beyond walking distance}. An illustrative example is in Table 2. Theoretically, we need strict monotonicity, because two categorical values differ to some extent. But, the user might indicate that for a particular case the difference between two specific neighbouring values is irrelevant.

Table 2
Proximity relation between ordinal categorical values expressed by numbers from the unit interval which satisfy the strict monotonicity.

s(x,y)	very low	low	medium	high	very high
very low	1	0.8	0.6	0.3	0
low		1	0.75	0.4	0.1
medium			1	0.6	0.3
high				1	0.8
very high					1

Definition 2. Let $I_s = \{1, \dots, n-1\}$ be an index set, in which n is the number of ordinal categorical values. Then monotonicity holds when for $\forall i, j \in I_s$ and $j > i$ we get $a_{ij} \ge a_{i,j+1}$, where $a_{ij} = s(x_i, x_j)$ and $a_{ij} \in [0,1]$.

Observe that, when $\ge$ is replaced by $>$, we get a strictly monotone proximity relation. Theoretically, we need strict monotonicity to keep the distinction between ordinal categorical data, but for some users and tasks, the difference between two neighbouring values might be irrelevant. Next, observe that this definition covers the min-min transitivity $s(x_i, x_j) = \min_{x_k \in LV} \min[s(x_i, x_k), s(x_k, x_j)]$.

An example of a proximity relation satisfying Definition 2 is in Table 2. It prevents from assigning values greater than 0.6 to *s*(*medium*, *very high*) because s(medium, high) = 0.6 and monotonicity holds. The same observation holds for the min-min transitivity. In this case, s(medium, high) = 0.6 and s(high, very high) = 0.8. From Definition 2, we recognize for the first row the relation: $0 \le s(a_{1,1}, a_{1,n}) < s(a_{1,1}, a_{1,n-1}) < \cdots < s(a_{1,1}, a_{1,2}) < s(a_{1,1}, a_{1,1}) = 1$.

Having proposed improvement, we move further in expressing proximity by the family of linguistic terms. For people, it is a more convenient way to express proximity. A possible drawback in comparison to the numerical proximity, is that we have an infinite set of real numbers in the unit interval and therefore we do not have a theoretical limitation for the number of categories and slight differences in proximities. On the other hand, people are not able to distinguish meanings and differences in terms inside a larger set of linguistic terms. According to the research by [45], the optimal number of categorical data should not be higher than seven ± two terms.

We need to adjust monotonicity to the set of linguistic terms. The extension of Definition 2 is the following

Definition 3. Let $I_s = \{1, \dots, n-1\}$ be an index set, in which n is the number of ordinal categorical values. Then monotonicity holds when for $\forall i, j \in I_s$ and $j > i$ we get $A_{ij} \succeq A_{i,j+1}$, where A_{ij} is a fuzzy set formalizing linguistic terms and $A_{ij} = s(x_i, x_j)$.

Observe that, when $\succeq$ is replaced by $\succ$, we get a strictly monotone proximity relation. An example of a family of linguistic terms expressed by TFNs is in Fig. 4. When we construct a family of fuzzy sets by applying fuzzy partition [23], the quality indicators raised in [24] and explored in [46], it simplifies further calculations. Instead of calculating the dominance measure or admissible orders, we compare the indexes of linguistic terms.

An example of linguistic proximity is in Table 3. The new property prevents the assignment of a linguistic term expressing closer similarly than *significant similarity* (term L_6 in Fig. 4) to s(medium, very high), as s(medium, high) is expressed by this term and the monotonicity holds. The same observation holds for min-min transitivity, in this case:

Table 3
Proximity relation between ordinal categorical values expressed by linguistic terms which satisfy the strict monotonicity of TFNs.

s(x,y)	very low	low	medium	high	very high
very low	same	almost same	more or less similar	significantly different	fully different
low		same	very similar	more or less similar	very different
medium			same	significantly similar	more or less similar
high				same	very similar
very high					same

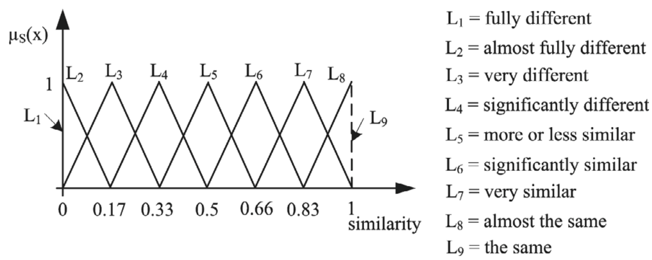

Fig. 4. A family of fuzzy numbers formalizing the proximity relation among categorical data.

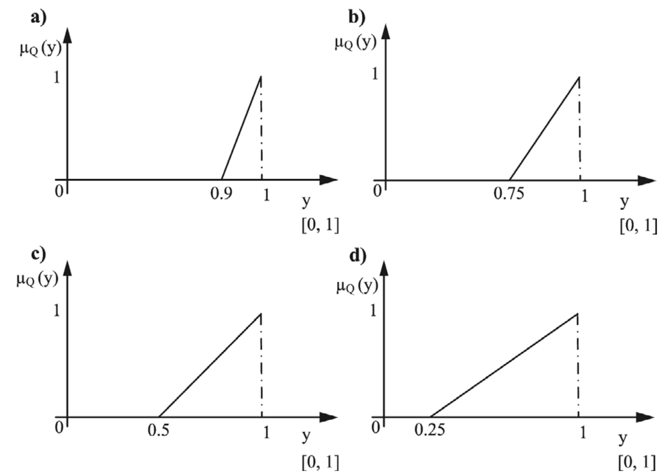

Fig. 5. Fuzzy relative quantifier *most of* from Fig. 3 adjusted to linguistic requirements: (a) almost all, (b) strong majority, (c) most of, and (d) at least several. In all cases, it is a strictly increasing function.

s(medium, high) has *significant similarity* (L_6) and s(high, very high) is *very similar* (L_7).

The conformance measure (8) requires numerical values. But *min* and *max* operations can be extended to fuzzy numbers by comparison operators [21]. Generally, to find whether a fuzzy number A_1 is smaller than a fuzzy number A_2, we might apply the admissible order (see, e.g., [47]).

When we have a well-defined family of linguistic terms (LV) expressing proximity, then it is a straightforward task to find a smaller TFN. By the well-defined family, we mean that the following criteria for TFN are met: normality, continuity, convexity, and unimodality. Next, the following criteria for linguistic variable are met: relation preservation, distinguishability, coverage, uniform granulation, leftmost/rightmost fuzzy set [24]. The majority of the requirements are also covered by fuzzy partitions defined in [23]. By direct applying the dominance relation (7), we see in Fig. 4 that, for instance, $L6$ dominates over $L5$, i.e., $\min\{L5, L6\} = L5$.

Example 1. For instance, let us have a user requirement u that the value should be *very low*, but the values for a record r are *low, medium*. The conformance is the following (Table 3 and Fig. 4):

$$
\begin{aligned}
&C(\text{attribute}[u, r]) = \\
&\min\{\min\{\max\{s(VL, L), s(VL, M)\}\}, \\
&\min\{\max\{s(L, VL)\}, \max\{s(M, VL)\}\}\} \\
&= \min\{\min\{\max\{L8, L5\}\}, \min\{\max\{L8\}, \max\{L5\}\}\} \\
&= \min\{\min\{L8\}, \min\{L8, L5\}\} = \min\{L8, L5\} = L5
\end{aligned}
$$

i.e., the conformance gets value *more or less similar*. In the same way, we can evaluate the conformances for the other records and considered attributes.

The conformances are either TFNs or crisp values. It means that we are able either to straightforwardly aggregate TFNs and crisp values (when transformed to TFN of support and core only in crisp value) using an extension of the appropriate aggregation function, or defuzzify TFNs before applying aggregation. In the former, we keep uncertainty through the whole process.

3.2. Quantified aggregation of atomic conformances

A high variety of functions for aggregating elementary conformances can be applied. The basic ones are: (weighted) arithmetic (or other) mean, conjunction, and disjunction. The strong point is their simplicity. On the other hand, they are oversimplified for more complex aggregations (Section 2.2). They benefits and weak points are advocated in, e.g., [11] and examined for recommender systems in [2]. In this section, quantified aggregation is explored.

Aggregation by the fuzzy relative quantifier *most of* (or its variants) evaluates whether the majority of conformances is significantly satisfied. The first question is adjusting quantifier to the user needs. When the user says all elementary requirements (expressed by conformances) should be significantly satisfied, then we apply a nilpotent conjunction (more about this function is in e.g., [10,38]). On the other hand, when the user says at least one elementary requirement should be significantly satisfied, then we apply a nilpotent disjunction. To cover less restrictive evaluation than conjunction and more restricted evaluation than disjunction, we apply quantified aggregation in the following way:

- When the user says *almost all* elementary requirements should be met, we assign values, e.g., $l = 0.9$ and $p = 1$ (Fig. 5a).
- When the user says *strong majority* of elementary requirements should be met, we assign values, e.g., $l = 0.75$ and $p = 1$ (Fig. 5b).
- When the user says *most of* elementary requirements should be met, we assign values, e.g., $l = 0.5$ and $p = 1$ (Fig. 5b).
- When the user says *at least several* elementary requirements should be met, we assign values, e.g., $l = 0.25$ and $p = 1$ (Fig. 5d).

Observe that we keep $p = 1$ (12). Our goal is to recommend the most suitable products or services. When several products are near the ideal satisfaction, then we should distinguish them and sort downward from the best by the membership degree of belonging to the quantifier *most of* (compare Figs. 3 and 5). The similar observation was recognized when interpreting quantified linguistic summaries expressing hierarchical data on maps [40].

The suggestion based on this aggregation method (and others) copes with the question of product (or service) prioritization in the case when more products (or services) get the same score. This problem is

Table 4
Two rooms and their elementary conformances to the raised requirements.

Conformance (8)	Room R1				Room R2			
	a	m	b	L_i [b]	a	m	b	L_i [b]
C1	0	0.17	0.33	L_3	0.66	0.83	1	L_7
C2	0.33	0.5	0.66	L_5	0.5	0.66	0.83	L_6
C3	0	0	0.17	L_2	0.83	1	1	L_8
C4	0.66	0.83	1	L_7	0	0	0	L_1
C5	0.17	0.33	0.5	L_4	1	1	1	L_9
C6	0.83	1	1	L_8	0.33	0.5	0.66	L_5
C7	0.17	0.33	0.5	L_4	1	1	1	L_9
C8	0.5	0.66	0.83	L_6	0.83	1	1	L_8
C9	0	0.17	0.33	L_3	0.66	0.83	1	L_7
C10 [a]		0.75				1		

[a] Light preference of balcony (a = 0.75) in Table 1, i.e., room *R1* does not have a balcony, room *R2* has balcony.
[b] Linguistic term and its values from Fig. 4.

Table 5
Result of recommendation by conformances and quantified aggregation, see Fig. 6.

Quantifier	Most of		Strong majority	
Room	Possibility (13)	Necessity (14)	Possibility	Necessity
R1	0.146	0	0	0
R2	0.616	0.403	0.304	0,071

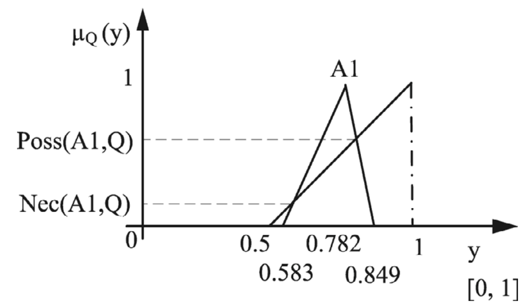

Fig. 6. Possibility and necessity values for TFN A_1 in the quantifier *most of*.

mitigated in quantified aggregation when we apply a strictly monotone increasing quantifier and TFNs in proximity functions. However, there might be two or more entities which satisfy requirements with the same highest degree. A simple solution is randomly selecting one, as there is no further information. A more advanced solution is the evaluation of elementary conformances. When a user poses a set of requirements, we might consider that the user raises them in the order of relevance. For instance, the user starts with the *distance to the beach* followed by the *number of stars*, we expect that the distance is a more relevant requirement than the number of stars. So, we compare the matching degrees of elementary conformances in the lexicographic order. It is similar like a decision tree proposed in [2].

Example 2. Let us have two hotel rooms which have conformances to the user requirements in Table 4. The linguistic interpretation of conformances reflects a hypothetical proximity relation which satisfies the quality measures of a family of fuzzy sets and Definition 3, to illustrate the application of possibility and necessity measures. For instance, the conformance *C8* for room *R1* is calculated in the way shown in Example 1.

By applying the sum of fuzzy numbers (3) from Table 4 and their multiplication by parameter $1/10$ (4), we get the fuzzy proportion (TFN A_1) for room *R1* as $y_{R1} = A_1(0.341, 0.472, 0.59)$ and TFN A_2 for room *R2* as $y_{R2} = A_2(0.583, 0.782, 0.849)$.

The next step is calculating the possibility (5) and necessity (6) that these two TFNs belong to the fuzzy quantifiers illustrated in Fig. 5. We apply the adapted possibility and necessity measures to fuzzy numbers [20] as follows:

$$Poss_{A,Q}(y) = \begin{cases} 1, & \text{for } m \geq p \\ 0, & \text{for } b \leq l \\ \frac{b-l}{(p-l)-(m-b)}, & \text{otherwise} \end{cases} \tag{13}$$

where the parameters of quantifier and fuzzy number are shown in Figs. 1 and 3.

Next, the necessity measure is expressed as:

$$Nec_{A,Q}(y) = \begin{cases} 1, & \text{for } a \geq p \\ 0, & \text{for } m \leq l \\ \frac{m-l}{(p-l)-(a-m)}, & \text{otherwise} \end{cases} \tag{14}$$

where the parameters of quantifier and fuzzy number are shown in Figs. 1 and 3.

The results are in Table 5 for quantifiers *most of* and *strong majority* (see Fig. 5c and b, respectively).

In this example, room *R2* is treated as a better option even though it fully dissatisfies conformance *C4*. It significantly better satisfies the majority of conformances in comparison to the room *R1*, although *R1* meets at least very weakly all conformances. In the case of conjunctive evaluation, room *R2* will be rejected, because 0 is an annihilator for each conjunctive function.

The next flexibility is applying the possibility or necessity measure in the final calculation. The former is an optimistic value, whereas the latter is a pessimistic value. They reflect a less demanding or a high-end user, respectively. Theoretically, it holds, but the necessity measure gets a value greater than 0, when the intersection of the cores of the fuzzy number and the support of the quantifier is not an empty set for a strictly increasing quantifier. The next way is a convex combination of the possibility and necessity measures formalized in the following form:

$$\mu(A,Q) = \lambda \cdot Pos(A,Q) + (1-\lambda)(Nec(A,Q)) \tag{15}$$

where $Pos(A,Q)$ is calculated by (5), $Nec(A,Q)$ is calculated by (6), and $\lambda \in [0,1]$. For $\lambda = 0.5$, we get the arithmetic mean of possibility and necessity measures, while for $\lambda = 1$ we get the possibility measure and analogously for $\lambda = 0$ we get the necessity measure. The convex combination covers a smooth transition from high-end to low-demanding customers.

For strictly increasing quantifiers, the maximal satisfaction of possibility and necessity holds only when the modal value m of the resulting TFN is equal to 1. Hence, the ideal possibility is assigned to an entity satisfying each conformance ideally.

Let us check the axioms of aggregation functions for the proposed quantified aggregation. The boundary conditions and monotonicity hold, regardless we apply the possibility or necessity measures as well as their convex combination. Clearly, if all conformances are expressed as L_9 (4), then the fuzzy proportion gets value of 1, which provides the ideal solution regardless which fuzzy quantifier (5) we apply. The opposite holds when all conformances are expressed by L_1. The proof of monotonicity is a matter of direct computation.

4. Experiments

In the experiment, hotel rooms for a holiday are evaluated. In our data set, we have generated data for 50 hotel rooms to avoid any advertisement. Next, the following ten attributes explain hotel rooms: A1: distance to the beach, A2: number of sunny days, A3: hotel size, A4: number of stars, A5: evaluation of quests, A6: sustainability support by hotel, A7: pollution, A8: length of fly (distance from the home country), A9: price, and A10: balcony.

In Table 6 all data are ordinal categorical for the purpose of the experiment. Categorical and binary data are directly applicable, while the numerical data should be categorized. The categorization is supported

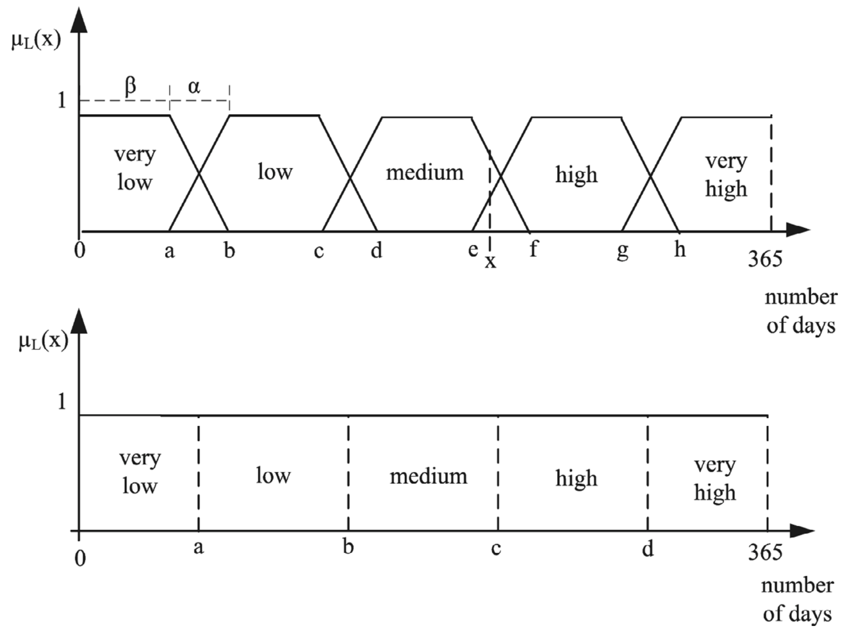

Fig. 7. Categorization of the attribute *the number of sunny days* by linguistic variable and categorical values.

by diverse ways (see, e.g., [36]): equi-length, equi-depth, equi-log (or any other function, or statistical methods based on data distribution). In the fuzzy sets environment, we apply linguistic variables and fuzzy partitions. An example of the attribute *the number of sunny days* is depicted in Fig. 7.

Clearly, when $\alpha = 0$, we cover the classical categories, while for $\beta = 0$, we handle the maximal uncertainty. Next, in the uncertain part between fuzzy sets ($\alpha > 0$), we assign either a categorical value with a higher membership degree (for value x it is *medium*) or adopt both neighbouring categories or linguistic terms (in this case {medium, high}). In the experiment, we assigned one categorical value (with a higher membership degree), which does not influence generality. When both values are assigned, then the conformance measure calculation (8) covers this case easily.

Customer requirements are in Table 7. To avoid any confusion with conformance or fuzzy sets, we have marked the customers as $U1$, $U2$, and $U3$. Customers express their requirements using a single linguistic term, or with uncertainty by two terms, e.g., very low, low. These three illustrative customers pose relatively similar elementary requirements to examine whether we obtain comparable results.

Customers posed an aggregated request that the majority of elementary requirements should be met. For all customers, we compared the results for quantifiers *most of*, *strong majority* and *almost all* (Fig. 5), i.e., from relaxed to a stricter aggregation.

Results are in Table 8. Customers $U1$ and $U2$ differ only in the requirement for balcony. Rooms $R40$ and $R41$ are selected with matching degrees around 0.6 for both customers, with a slight preference on $R40$. Customer $U2$ has a bit stronger requirement over this binary data, while both rooms are very similar considering all attributes. In the case of binary elementary requirement, both rooms will be fully rejected. Room $R40$ is on the fourth position. If the three best rooms are occupied, this room is recommended as an alternative without further calculations. This room also partially passes a more strict quantifier *strong majority* for both customers. This room is fully rejected only for quantifier *almost all* for customer $U2$. For both customers, the recommended room is $R45$, except when the aggregation is very strict, which causes that the winner is $R44$.

Customer $U3$ poses several stricter requirements and therefore less rooms meet the requirement for all three quantifiers. It causes that the winner is room $R40$. In the case of classical queries or conjunctive aggregation, the two similar rooms $R44$ and $R45$, might be differently treated.

In our experiment, attribute *A9* is price. Room *R38* has a very high price. It is a very strict restriction for customers *U1* and *U2* but for customer *U3* it means rejection. When applied to the usual quantified aggregation (11), this room is among the considered ones for fuzzy quantifier *most of* (see Table 8). However, with a mandatory requirement for price according to Eq. (17) this room is fully rejected.

The next option is applying the necessity measure (6) or the convex combination of possibility and necessity measures (15).

5. Discussion

Proximity relations are usually subjective and therefore different values can lead to differences in results. Selecting the appropriate similarity between categorical values (i.e., is the similarity between *very low* and *low* the same as between *high* and *very high*?) is crucial to ensure meaningful recommendation. Today, when evaluating alternatives, people consider environmental requirements such as sustainability support and pollution. Usually, these requirements are not mandatory but might influence ranking. It is not an easy task for people to express these requirements using sharp criteria. Each alternative has a financial aspect expressed by its price related to the other features. A sharp requirement for price might cause that an alternative which is very slightly beyond the border, excellently satisfies the other features. A relaxation of price requirement might emphasize such an alternative, but not fully. In this case, the solution is a conjunctive aggregation of the degree of matching to the acceptable price with the quantified aggregation of the other requirements.

In the traditional way, the elementary conformances get values from the unit interval (i.e., [2,9,48]), which opens the space for applying all aggregation functions. On the other hand, thanks to the extension principle [18,19], even conformances expressed by TFNs on the unit interval can be seamlessly aggregated by any logic aggregation function.

The user's feeling in assigning values from the unit interval is disputable. As was mentioned earlier, the initial definition of the proximity relation by [8] is sufficient for nominal categorical data but lacks the consistency of the proximity matrix for ordinal ones. In this case, the user should be careful when assigning proximity values, which

Table 6
Hotel rooms and values of their attributes.

Room	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
R1	high	low	medium	low	low	high	v low	medium	low	yes
R2	v high	v high	low	medium	v high	high	medium	v low	high	yes
R3	low	high	v high	high	high	medium	low	v high	low	no
R4	medium	v low	high	low	medium	v high	high	v high	medium	no
R5	medium	high	v low	v high	v high	high	high	medium	low	no
R6	medium	medium	v low	low	medium	low	high	low	v low	yes
R7	high	v low	v low	v high	medium	high	high	high	v high	no
R8	high	v high	v low	low	v high	medium	high	v high	v high	no
R9	low	high	v high	low	v low	high	low	v high	high	no
R10	v low	v high	medium	low	medium	medium	low	v high	medium	yes
R11	v high	v low	v high	high	v high	low	low	v high	medium	yes
R12	v low	v high	v high	low	medium	v high	low	medium	medium	no
R13	low	v high	medium	v low	low	v high	v low	medium	v low	no
R14	v high	high	high	v high	high	v low	low	medium	v high	no
R15	v low	medium	high	low	v high	medium	high	medium	low	yes
R16	high	v high	v high	low	medium	low	medium	low	medium	yes
R17	medium	medium	medium	medium	v low	low	v low	v high	low	yes
R18	low	v high	v high	v high	v high	v high	v low	v high	low	yes
R19	v low	v low	v low	v low	low	high	v high	high	high	yes
R20	medium	low	medium	medium	medium	v high	medium	low	v high	no
R21	medium	low	low	medium	v high	v high	medium	v high	v low	yes
R22	high	medium	high	v high	high	high	low	medium	high	yes
R23	v low	high	v low	high	medium	v high	low	v high	v low	yes
R24	low	high	low	high	medium	v high	high	v high	high	no
R25	v low	low	high	low	v high	low	low	v high	low	yes
R26	v low	v high	v high	low	high	medium	v low	high	v low	no
R27	v low	medium	low	high	low	medium	high	high	v high	yes
R28	v high	low	v high	v low	low	medium	v high	high	v low	yes
R29	medium	v high	medium	v low	low	low	v high	v low	high	no
R30	v high	high	high	v high	medium	low	high	high	v low	no
R31	high	medium	v high	medium	high	low	medium	v low	high	yes
R32	medium	medium	medium	medium	v low	v low	v high	v low	v high	no
R33	v high	high	high	medium	medium	low	low	v low	low	yes
R34	v high	v high	medium	v low	high	medium	high	low	low	no
R35	low	low	high	medium	high	low	v low	v low	low	yes
R36	v low	medium	high	v low	low	medium	low	high	medium	no
R37	medium	v low	v low	v low	medium	v low	v high	high	low	yes
R38	high	high	v high	high	high	v high	high	high	v high	no
R39	v high	v low	v high	v low	high	high	v high	high	high	no
R40	v high	v high	high	v high	v high	high	v high	low	low	no
R41	v high	high	v high	v high	v high	high	v high	v low	v low	no
R42	high	v high	v high	v high	high	high	high	low	low	yes
R43	v high	high	high	high	v high	high	v high	medium	v low	no
R44	high	high	v high	v high	v high	v high	high	v low	v low	yes
R45	high	high	high	v high	high	high	v high	low	medium	yes
R46	v low	medium	low	medium	medium	medium	low	v high	v high	no
R47	low	v low	v low	v low	medium	v low	v low	v high	v high	yes
R48	low	low	low	v low	v low	low	v low	v high	v high	yes
R49	low	v low	low	low	medium	medium	medium	v high	v high	no
R50	v low	v low	v low	medium	low	low	v low	high	v high	no

Table 7
Elementary requirements posed by customers.

Customer	A1	A2	A3	A4	A5
Customer U1	{high, very high}	high	{medium, high}	high	medium
Customer U2	{high, very high}	high	{medium, high}	high	medium
Customer U3	{high, very high}	very high	{medium, low}	very high	medium

Customer	A6	A7	A8	A9	A10
Customer U1	very high	{high, very high}	{very low, low}	low	light preference
Customer U2	very high	{high, very high}	{very low, low}	low	strong preference
Customer U3	very high	{high, very high}	very low	very low	just prefer

is not an easy task. However, for machine learning, it is a problem. It should have clear boundaries and mathematical rules for possible values to avoid under or over-fitting. In essence, we have enhanced the proximity relation in two directions. The first improvement involves introducing the property of monotonicity for ordinal categorical data, thereby incorporating a min-min transitivity to avoid inconsistencies in proximity matrices. The second contribution involves extending the proximity relation to handle the linguistical representation of similarity. Fuzzy numbers are natural means to formalize proximity relations in such scenarios. We also proposed satisfying the Ruspini partition (see Fig. 4) to simplify checking the monotonicity of fuzzy numbers. We created a conformance measure based on a proximity relation handled by triangular fuzzy numbers to create a uniform way for computing the similarity between data and user elementary requirements covering diverse data types. By proposing this, we bridge the research gap in expressing proximity through a consistent matrix for numerical and for linguistically expressed similarities among categorical as well as binary data.

Table 8
Rooms ranked for all three customers by the possibility measure (13).

Room	Mo[a] U1	St[a] U1	Al[a] U1	Mo U2	St U2	Al U2	Mo U3	St U3	Al U3
R45	0.753	0.585	0.301	0.753	0.585	0.301	0.673	0.450	0.069
R44	0.741	0.547	0.175	0.741	0.547	0.175	0.684	0.447	0
R42	0.727	0.541	0.225	0.727	0.541	0.225	0.712	0.497	0.087
R40	0.686	0.473	0.110	0.605	0.337	0	0.655	0.396	0
R41	0.678	0.449	0.035	0.595	0.306	0	0.582	0.283	0
R43	0.650	0.400	0	0.567	0.257	0	0.550	0.256	0
R34	0.578	0.291	0	0.497	0.155	0	0.512	0.147	0
R33	0.569	0.246	0	0.569	0.246	0	0.498	0.138	0
R30	0.567	0.259	0	0.484	0.117	0	0.498	0.140	0
R2	0.549	0.254	0	0.549	0.254	0	0.565	0.268	0
R16	0.538	0.222	0	0.538	0.222	0	0.387	0	0
R5	0.524	0.199	0	0.443	0.063	0	0.511	0.175	0
R38	0.512	0.163	0	0.428	0.020	0	0.350	0	0
R6	0.497	0.138	0	0.497	0.138	0	0.442	0.040	0
R22	0.483	0.160	0	0.483	0.160	0	0.418	0.037	0
R28	0.471	0.125	0	0.471	0.125	0	0.295	0	0
R1	0.456	0.085	0	0.456	0.085	0	0.377	0	0
R20	0.455	0.063	0	0.371	0	0	0.312	0	0
R31	0.443	0.079	0	0.443	0.079	0	0.312	0	0
R29	0.431	0.060	0	0.352	0	0	0.332	0	0
R24	0.425	0	0	0.336	0	0	0.388	0	0
R7	0.406	0.021	0	0.328	0	0	0.284	0	0
R4	0.400	0	0	0.317	0	0	0.139	0	0
R15	0.374	0	0	0.374	0	0	0.266	0	0
R39	0.369	0	0	0.292	0	0	0.107	0	0
R21	0.357	0	0	0.357	0	0	0.330	0	0
R13	0.345	0	0	0.262	0	0	0.263	0	0
R12	0.344	0	0	0.261	0	0	0.140	0	0
R23	0.343	0	0	0.343	0	0	0.352	0	0
R27	0.321	0	0	0.321	0	0	0.295	0	0
R32	0.307	0	0	0.226	0	0	0.191	0	0
R35	0.302	0	0	0.302	0	0	0.191	0	0
R18	0.284	0	0	0.284	0	0	0.241	0	0
R14	0.281	0	0	0.200	0	0	0.164	0	0
R37	0.276	0	0	0.276	0	0	0.145	0	0
R3	0.269	0	0	0.182	0	0	0.159	0	0
R19	0.261	0	0	0.261	0	0	0.110	0	0
R10	0.247	0	0	0.247	0	0	0.169	0	0
R8	0.247	0	0	0.168	0	0	0.135	0	0
R11	0.241	0	0	0.241	0	0	0.050	0	0
R36	0.221	0	0	0.142	0	0	0	0	0
R17	0.190	0	0	0.190	0	0	0.162	0	0
R26	0.177	0	0	0.093	0	0	0.026	0	0
R9	0.171	0	0	0.090	0	0	0	0	0
R49	0.090	0	0	0.008	0	0	0	0	0
R25	0.079	0	0	0.079	0	0	0	0	0

[a] Mo stands for quantifier *most of*, St for quantifier *strong majority* and Al for quantifier *almost all*.

The maximal number of possible different numerical proximity values s_p is

$$s_p = n + \sum_{i=2}^{n} (n - i) \tag{16}$$

where n is the number of categorical values (see Tables 2 and 3).

Thus, for five categorical values, we get 11 possible proximity values, while for six categories we get 16 possible linguistic values (note that a part of the matrix is symmetric, due to the property of reflexivity, and we count value 1 at the main diagonal only once). Theoretically, the same holds for proximities expressed linguistically. But humans are generally able to recognize linguistic terms that express concepts like proximity within the range seven ± two, that is, nine is usually the upper limit for the cognitive processing of categorical information [49].

The next step involves determining how to proceed with the aggregation of elementary crisp or fuzzy conformances. In our work, quantified aggregation is improved for the cases when the conjunctive behaviour is too restrictive, disjunctive to relaxing and averaging does not match with the expected results (see Section 2.2). Thus, our approach handles aggregation *to more to merrier*, which is one of the natural human aggregations. Next, when aggregating fuzzy numbers, either possibility (optimistic) or necessity (pessimistic) measures are applied. Thus, the former assigns a higher matching degree, while the latter due to its restrictiveness, might lead in many cases to degree equal to 0. A convex combination of these two measures is the solution. In this case, the necessity measure is applied for the high-end customers, while the possibility measure covers the non-demanding ones, and the adapted parameter $\lambda \in]0, 1[$ (15) handles customers between these two opposite cases.

The quantified aggregation does not support mandatory requirements. When only one requirement is fully rejected, while the other are significantly met, an entity is a feasible solution. It is problematic when one (or several) of the requirements should be at least partially met, e.g., *low price*. We cannot afford an expensive hotel regardless the other requirements are nicely met. It that case, the quantified aggregation is straightforwardly extended by the conjunction as follows [12]:

$$v(x) = (\bigwedge_{i=1}^{r} P_i(x)) \bigwedge \mu_Q(\frac{1}{s} \sum_{j=1}^{s} \mu_{P_j}(x)) \tag{17}$$

where r is the number of mandatory elementary requirements and s is the number of quantified elementary requirements (generally $r < s$), and x is a product or service.

The next straightforward extension is an asymmetric aggregation, where the user poses a question like *most of* requirements $P_i, i =$

$1, \ldots, m$ or else *most of* requirements $P_j, j = 1, \ldots, n$ should be met. It is expressed in the structure (most of $\sum_{i=1}^{m} P_i) \otimes$ (most of $\sum_{j=1}^{n} P_j$) is satisfied, where $\otimes$ expresses *or else* operator or *and if possible*. More about asymmetric conjunction and disjunction is in, e.g., [50, 51]. Due to the extension principle of fuzzy sets and a partial order of the well-defined family of TFNs (see Fig. 4), the computation is straightforwardly extended.

By adding monotonicity and flexibility to the proximity relation and conformance measure as well as the ability to aggregate fuzzy conformances using fuzzy quantifiers, the proposed model for a fuzzy query engine offers a novel approach.

The results obtained show that the proposed approach provides several advantages and features to enhance the flexibility and effectiveness of querying in scenarios where imprecise and approximate matching is essential. The main contributions pertain to the development of the adapted conformance measure for computing similarity using linguistic terms, the introduction of an extended proximity relation with new properties, and quantified aggregation of conformance values supported by the convex combination of possibility and necessity measures. It is important to note that while the existing fuzzy query approaches offer advantages in certain contexts, they may not be suitable for all applications.

Although proximity relations are valuable, they also have drawbacks. Computational complexity may be overcome with triangular fuzzy numbers proposed in this paper, but limitations such as a lack of semantic understanding, sensitivity to outliers in numeric domains when creating categories, and challenges in handling a larger number of categorical values have been recognized.

In this work, all functions expressing TFNs and quantifiers are linear, which may not be an ideal solution for all cases. For example, when a user specifies a quite significant majority or a quite weak majority, nonlinear functions might be a better option. Linear functions for proximities are less suitable when users hesitate to give their preferences for proximities. In this case, the proximities can be better represented by pi-shaped fuzzy numbers. They should also meet the Ruspini partition. This adjustment does not affect the model. The difference is only in the calculated matching degrees. Nevertheless, this is a topic for future research.

The next limitation is related to the aggregation of elementary requirements when their so-called coalitions exist. For example, the presence of a balcony might be more relevant when the room faces a see view and less relevant in the opposite case. The solution is fuzzy capacities and the Choquet integral [52]. However, computing all capacities can be a demanding task for end users. Coalitions might be expressed as subadditive (e.g., reducing relevancies of the simultaneously satisfied several elementary requirements) and superadditive (e.g., increasing relevancies of simultaneously satisfied several elementary requirements) [10]. Generally, it is a hard task for the user to assign all capacities (if we have 10 elementary requirements, then we get 100 capacities only for coalitions of two elementary conditions, plus collations of three, four, and so on, resulting in a complex task).

Further weaknesses of the proposed method are: (i) it is not able to learn from the experience, because this approach does not work with the purchase history. If this history is recorded, then it should have a quite complex structure for learning parameters of linguistic terms for proximities and for fuzzy relative quantifiers; (ii) missing values reduce the similarity score and aggregation; (iii) when applied on numerical attributes, it depends on the constructed discretization by fuzzy sets. On the other hand, non-existence of recorded customers' opinions reduces requirements for filtering inadequate data (e.g., reviews promoting or demoting specific products or services), [53], noise [54], and covering problems related to the cold start and sparse data.

Future work should focus on these weaknesses. For instance, the problem of missing values can be mitigated by estimation or by penalization. A value might be missing because it is simply not collected, or the data owner (e.g., hotel) has not provided it, because the value is not good for the owner. Nevertheless, in quantified aggregation if for one non-mandatory requirement we do not have data, we might consider it as a clear non-match. If the other requirements are significantly met, the record is not rejected. The problem of discretization of numeric variables to crisp or fuzzy partitions is context dependent and therefore it is not an easy task, especially for a less-frequently bought products or services and when users are not willing to open their personal data.

6. Conclusion

By adding flexibility to the proximity relation and conformance measure for similarity matching, as well as the ability to aggregate fuzzy conformances using fuzzy quantifiers, the suggested model for a fuzzy query engine offers a novel approach. When expressing their requirements for a product or service, customers prefer linguistic terms, a vague, but convenient way. To meet this requirement, we proposed a new property for the proximity relation: monotonicity, which is natural for humans, but should be considered when values are learnt by machines. Next, we express the proximity relation by linguistic terms handled by TFNs and therefore we get the fuzzy conformances of elementary requirements. For their quantified aggregation, the possibility measure is applied for the non-demanding customers, while the necessity measure covers the high-end ones, and their convex combination for customers between these two opposite cases. Nowadays, when evaluating alternatives, people also consider features like the support for sustainability and pollution without raising sharp boundaries. Usually, these requirements are not mandatory, but influence ranking. Each alternative has a financial aspect expressed by its price. A sharp requirement for price might cause that an alternative which is very slightly beyond the border, excellently meets the other features. A relaxation might emphasize such an alternative, but not fully.

We emphasize that our work is open for extensions by other aggregation functions (aggregating mandatory requirements with quantified aggregation, symmetric and non-symmetric functions, and ordinal sums of conjunctive and disjunctive functions for covering downward (less-preferred) and upward reinforcement (more-preferred) products, etc.) to create a more robust query engine.

While proximity relations are valuable for similarity measure tasks, they also have some limitations. All functions expressing triangular fuzzy numbers and quantifiers are linear, which may not be an ideal solution for all cases, e.g., when users hesitate in answers. In this case, the proximities can be better represented by pi-shaped fuzzy numbers. They should also meet the Ruspini partition. The next option is applying type-2 fuzzy sets or interval-valued fuzzy sets to handle the uncertainty and hesitance related to constructing fuzzy sets on numeric attributes. The limitation of aggregating elementary requirements in quantified aggregation is when so-called coalitions among features exist (e.g., the presence of a balcony might be more relevant when the room faces see view and less relevant in the opposite case). Nevertheless, these limitations raise topics for future research.

Our approach does not have the ability to learn from the experience, because the proposed approach does not work with the purchase history. If such data are available, the learning model should be adapted to work with a higher number of parameters handling proximity terms and quantified aggregation. On the other hand, the proposed properties for proximity relation and families of linguistic terms which are Ruspini fuzzy partitions, reduce the complexity for learning parameters to the subsets of possible values and their relations. The same holds for assigning parameter to the convex combination of possibility and necessity measures.

CRediT authorship contribution statement

Miroslav Hudec: Writing – original draft, Supervision, Methodology, Formal analysis, Conceptualization. **Miljan Vučetić:** Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Nina Barčáková:** Validation, Methodology, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

This article has been produced with the financial support of the European Union under the REFRESH – Research Excellence For REgion Sustainability and High-tech Industries project number CZ.10.03.01/00/22_003/0000048 via the Operational Programme Just Transition. Also, this article is based upon work from the COST Action CA19130, FinAI - Fintech and Artificial Intelligence in Finance - Towards a transparent financial industry, supported by COST (European Cooperation in Science and Technology). The partial support of the SGS project No. SP2022/113 of the Ministry of Education, Youth and Sports of the Czech Republic is kindly appreciated. Finally, the authors would like to express their sincere appreciation to Vlatacom Institute for its support of this research.

References

[1] A. Da Costa, M. Manzato, Exploiting multimodal interactions in recommender systems with ensemble algorithms, Inf. Syst. 56 (2016) 120–132.
[2] M. Vučetić, M. Hudec, A fuzzy query engine for suggesting the products based on conformance and asymmetric conjunction, Expert Syst. Appl. 101 (2018) 143–158.
[3] R. Viertl, Statistical Methods for Fuzzy Data, John Wiley & Sons, Chichester, 2011.
[4] L. Donzé, A. Meier, in: A. Meier, L. Donzé (Eds.), Applying Fuzzy Logic and Fuzzy Methods to Marketing, in: Fuzzy Methods for Customer Relationship Management and Marketing, Business Science Reference, Hershey, 2012, pp. 1–14.
[5] A. Skowron, A. Jankowski, R. Swiniarski, in: J. Kacprzyk, W. Pedrycz (Eds.), Foundations of Rough Sets, in: Handbook of Computational Intelligence, Springer, Berlin Heidelberg, 2015, pp. 331–348.
[6] L.A. Zadeh, Fuzzy sets, Inf. Control 8 (1965) 338–353.
[7] H. Zimmermann, Fuzzy Set Theory – and Its Applications, Kluwer Academic Publishers, London, 2001.
[8] S. Shenoi, A. Melton, Proximity relations in the fuzzy relational database model, Fuzzy Sets and Systems 100 (1999) 51–62.
[9] M.I. Sözat, A. Yazici, A complete axiomatization for fuzzy functional and multivalued dependencies in fuzzy database relations, Fuzzy Sets and Systems 117 (2001) 161–181.
[10] G. Beliakov, A. Pradera, T. Calvo, Aggregation Functions: A Guide for Practitioners, Springer-Verlag, Berlin Heidelberg, 2007.
[11] J. Dujmović, Soft Computing Evaluation Logic: The LSP Decision Method and Its Applications, Wiley–IEEE Computer Society, Hoboken, 2018.
[12] P. Sojka, M. Hudec, M. Švaňa, Linguistic summaries in evaluating elementary conditions, summarizing data and managing nested queries, Informatica 31 (2020) 841–856.
[13] V. Torra, Y. Narukawa, Modeling Decisions. Information Fusion and Aggregation Operators, Springer, Heidelberg, 2007.
[14] M. Vučetić, Z. Brokešová, M. Hudec, E. Pastoráková, Financial literacy and psychological disaster preparedness: applicability of approach based on fuzzy functional dependencies, Inf. Process. Manage. 59 (2022) id. 102848.
[15] M. Vučetić, M. Hudec, B. Božilović, Fuzzy functional dependencies and linguistic interpretations employed in knowledge discovery tasks from relational databases, Eng. Appl. Artif. Intell. 88 (2020) id. 103395.
[16] A. Kechagias, B. Papadopoulos, Computational method to evaluate fuzzy arithmetic operations, Appl. Math. Comput. 185 (1) (2007) 169–177.
[17] N. Seresht, A. Fayek, Computational method for fuzzy arithmetic operations on triangular fuzzy numbers by extension principle, Internat. J. Approx. Reason. 106 (2019) 172–193.
[18] J. Ramík, M. Vlach, Generalized Concavity in Fuzzy Optimization and Decision Analysis, Springer, Cham, 2012.
[19] L.A. Zadeh, Fuzzy logic=computing with words, Fuzzy Sets Syst. 4 (1996) 103–111.
[20] J. Galindo, A. Urrutia, M. Piattini, Fuzzy Databases – Modelling, Design and Implementation, Idea Group Publishing, Hershey, 2006.
[21] X. Wang, D. Ruan, E. Kerre, Mathematics of Fuzziness — Basic Issues, Springer, Berlin Heidelberg, 2009.
[22] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning: Part I, Inform. Sci. 8 (1975) 199–249.
[23] E. Ruspini, A new approach to clustering, Inf. Control 15 (1969) 22–32.
[24] J.M. Alonso, C. Castiello, L. Magdalena, C. Mencar, Explainable Fuzzy Systems: Paving the way from Interpretable Fuzzy Systems to Explainable AI Systems, Springer, Cham, 2021.
[25] S. Zadrożny, G. de Tré, R. de Caluwe, J. Kacprzyk, in: J. Galindo (Ed.), An Overview of Fuzzy Approaches to Flexible Database Querying, in: Handbook of Research on Fuzzy Information Processing in Databases, Information Science Reference, Hershey, 2008, pp. 34–55.
[26] S. Massanet, J. Recasens, J. Torrens, Fuzzy implication functions based on powers of continuous t-norms, Internat. J. Approx. Reason. 83 (2017) 265–279.
[27] G. Smits, M. Lesot, O. Pivert, R. Yager, Flexible querying using disjunctive concepts, in: Proceedings of the Flexible Query Answering Systems, FQAS 2021, Bratislava, 2021, pp. 1–2.
[28] M. Hudec, M. Vučetić‘, Some issues of fuzzy querying in relational databases, Kybernetika 51 (6) (2015) 994–1022.
[29] P. Bosc, A. Hadjali, O. Pivert, Empty versus overabundant answers to flexible relational queries, Fuzzy Sets and Systems 159 (2008) 1450–1467.
[30] G. Smits, M. Lesot, O. Pivert, M. Reformat, Diversifying top-k answers in a query by example setting, in: Proceedings of the Flexible Query Answering Systems, FQAS 2023, Palma, 2023, pp. 1–2.
[31] L. Caruccio, V. Deufemia, G. Polese, Relaxed functional dependencies —A survey of approaches, IEEE Trans. Knowl. Data Eng. 28 (2016) 147–165.
[32] C. Date, Date on databases: Writings 2000 – 2006, Apress, New York, 2006.
[33] F. Petry, Fuzzy Databases, Principles and Applications, Kluwer Academic Publishers, Boston, 1996.
[34] B. Buckley, F. Petry, A fuzzy representation of data for relational databases, Fuzzy Sets and Systems 7 (1982) 213–226.
[35] J. Peckol, Introduction to Fuzzy Logic, first ed., John Wiley & Sons, Hoboken, 2021.
[36] C. Aggrawal, Data Mining: The Textbook, Springer, Cham, 2015.
[37] D. Dubois, H. Prade, On the use of aggregation operations in information fusion processes, Fuzzy Sets and Systems 142 (2004) 143–161.
[38] M. Grabisch, J.-L. Marichal, R. Mesiar, E. Pap, Aggregation Functions, in: Encyclopedia of Mathematics and its Applications, No. 127, Cambridge University Press, Cambridge, 2009.
[39] G. Beliakov, H. Bustince, T. Calvo, A Practical Guide to Averaging Functions, Springer, Berlin, 2016.
[40] M. Hudec, K. Malovcová, R. Trumic, E. Rakovská, A new quality measure and visualization of the short-quantified sentences of natural language on maps – a case on COVID-19 data, Informatica 33 (2022) 321–342.
[41] M. Hudec, E. Mináriková, R. Mesiar, A. Saranti, A. Holzinger, Classification by ordinal sums of conjunctive and disjunctive functions for explainable AI and interpretable machine learning solution, Knowl.-Based Syst. 220 (2021) 106916.
[42] J. Kacprzyk, A. Ziółkowski, Database queries with fuzzy linguistic quantifiers, IEEE Trans. Syst. Man Cybern. SMC-16(3) (1986) 474–479.
[43] J. Kacprzyk, S. Zadrożny, Protoforms of linguistic database summaries as a human consistent tool for using natural language in data mining, Int. J. Softw. Sci. Comput. Intell. 1 (2005) 1–11.
[44] M.-J. Lesot, G. Moyse, B. Bouchon-Meunier, Interpretability of fuzzy linguistic summaries, Fuzzy Sets and Systems 292x (2016) 307–317.
[45] G. Miller, The magical number seven, plus or minus two: Some limits on our capacity for processing information, Psychol. Rev. 63 (1956) 81–97.
[46] E. Mináriková, Criteria for fuzzy-rule based systems and its applications on examples, in: Proceedings 24th Conference for Doctoral Students and Post-Doctoral Scholars, EDAMBA, Bratislava, 2021, pp. 1–2.
[47] L. De Miguel, H. Bustince, H. Fernandez, E. Induráin, A. Kolesárová, R. Mesiar, Construction of admissible linear orders for interval-valued atanassov intuitionistic fuzzy sets with an application to decision making, Inf. Fusion 27 (2016) 189–197.
[48] M. Vučetić, M. Hudec, M. Vujošević, A new method for computing fuzzy functional dependencies in relational database systems, Expert Syst. Appl. 40 (7) (2013) 2738–2745.
[49] G. Miller, The magical number seven, plus or minus two: Some limits on our capacity for processing information, Psychol. Rev. 63 (2) (1956) 81–97.
[50] M. Hudec, R. Mesiar, The axiomatization of asymmetric disjunction and conjunction, Inf. Fusion 53 (2020) 165–173.
[51] P. Bosc, O. Pivert, On four noncommutative fuzzy connectives and their axiomatization, Fuzzy Sets and Systems 202 (2012) 42–60.
[52] Z. Takáč, M. Uriz, M. Galar, D. Paternain, H. Bustince, Discrete IV dG-Choquet integrals with respect to admissible orders, Fuzzy Sets and Systems 441 (3) (2021) 169–195.
[53] V. Anelli, Y. Deldjoo, T. DiNoia, F. Merra, in: F. Ricci, L. Rokach, B. Shapira (Eds.), Adversarial Recommender Systems: Attack, Defense, and Advances, in: Recommender Systems Handbook, Springer US, New York, 2022, pp. 335–379.
[54] R. Yera, J. Castro, L. Martínez, in: G. Acampora, W. Pedrycz, A. Vasilakos, A. Vitiello (Eds.), Natural Noise Management in Recommender Systems Using Fuzzy Tools, in: Computational Intelligence for Semantic Knowledge Management: New Perspectives for Designing and Organizing Information Systems, Springer International Publishing, Cham, 2020, pp. 1–24.