Int. J. Appl. Math. Comput. Sci., 2000, Vol.10, No.4, 813–834

A FUZZY LOGIC BASED APPROACH TO LINGUISTIC SUMMARIES OF DATABASES

JANUSZ KACPRZYK*, RONALD R. YAGER**
SŁAWOMIR ZADROŻNY*

In this paper, we present basic ideas and perspectives related to the use of fuzzy logic for the derivation of linguistic summaries of data (databases). We concentrate on the issue of how to measure the goodness of a linguistic summary, and on how to embed data summarization within the fuzzy querying environment, for an effective and efficient implementation. In particular, we propose how to efficiently implement Kacprzyk and Yager's (2000) new quality indicators of linguistic summaries to derive summaries via Kacprzyk and Zadrożny's (1994; 1995a; 1995b; 1996) fuzzy querying add-on. Finally, we present an implementation for deriving linguistic summaries of a sales database at a computer retailer, and show how the linguistic summaries obtained can be useful for supporting decisions of the business owner.

Keywords: fuzzy logic, linguistic summary, computing with words, data mining, fuzzy querying

1. Introduction

The recent growth of Information Technology (IT) has implied, among others, the availability of a huge amount of data (from diverse, often remote databases). Unfortunately, the raw data alone are often not useful and do not provide 'knowledge'. More important are relevant, nontrivial dependencies which are encoded in those data. Unfortunately, they are usually hidden, and their discovery is not a trivial act, and requires some intelligence.

One of interesting and promising approaches to discover such dependencies is to derive linguistic summaries of a set of data (a database). Here we discuss linguistic summarization of data sets in the sense of Yager (1982; 1989; 1991; 1994; 1995a; 1995b; 1996) (for some extensions and other related issues, see, e.g., Kacprzyk and Yager, 2000; Rasmussen and Yager, 1996; 1997a; 1997b; 1999; Yager and Rubinson 1981) in which linguistic summaries are derived as linguistically quantified propositions,

* Polish Academy of Sciences, Systems Research Institute, ul. Newelska 6, 01–447 Warsaw, Poland, e-mail: `kacprzyk@ibspan.waw.pl`

** Machine Intelligence Institute, Iona College, New Rochelle, NY 10801, USA, e-mail: `yager@panix.com`

exemplified by—when the data in question concern employees—'most of the employees are young and well paid', with which a degree of validity is associated.

Originally, that degree of validity was meant to be the degree of truth of a linguistically quantified proposition that constitutes a summary. This was shown to be not enough, and other validity (quality) indicators were proposed. We mention George and Srikanth (1996), the solution of in which a compromise between the specificity and generality of a summary is sought, and then present some extension of Kacprzyk and Strykowski's (1999a; 1999b) approach in which a weighted sum of 5 quality indicators is employed that was proposed by Kacprzyk and Yager (2000).

We follow Kacprzyk and Zadrożny's (1998; 1999; 2000a; 2000b; 2000c), Kacprzyk's (1999a), and Zadrożny and Kacprzyk's (1999) idea of an interactive approach to linguistic summaries in which, since a fully automatic generation of linguistic summaries is not feasible at present, an interaction with the user is assumed for the determination of a class of summaries of interest via Kacprzyk and Zadrożny's (1994; 1995a; 1995b; 1996) fuzzy querying add-on to Microsoft Access.

We propose how to efficiently implement Kacprzyk and Yager's (2000) new quality indicators of linguistic summaries to derive summaries via Kacprzyk and Zadrożny's (1994; 1995a; 1995b; 1996) fuzzy querying add-on, and following Kacprzyk and Zadrożny's (1998; 2000a; 2000b; 2000c) idea of an interactive approach to linguistic summaries mentioned above.

We present an implementation of the proposed approach to the derivation of linguistic summaries for a sales database of a computer retailer. We show that the linguistic summaries obtained can be very useful for supporting decision making by the owner related to some relevant aspects of business functioning and running.

2. Linguistic Summaries Using Fuzzy Logic with Linguistic Quantifiers—the Basic Case

Here we will briefly present some basics of Yager's (1982) approach to the linguistic summarization of sets of data. This will provide a point of departure for our further analysis of more complicated and realistic summaries.

In Yager's (1982) approach, we have:

- V is a quality (attribute) of interest, e.g., the salary in a database of workers,
- $Y = \{y_1, \dots, y_n\}$ is a set of objects (records) that manifest quality V, e.g., the set of workers; hence $V(y_i)$ are values of quality V for object y_i,
- $D = \{V(y_1), \dots, V(y_n)\}$ is a set of data (the 'database' in question)

A *summary* of a data set consists of:

- a summarizer S (e.g., young),
- a quantity in agreement Q (e.g., most),
- truth (validity) T, e.g., 0.7,

as, e.g., 'T(*most* of employees are *young*)$= 0.7$'.

Basically, given a set of data D, we can hypothetize any appropriate summarizer S and any quantity in agreement, Q, and the assumed measure of truth (validity) will indicate the truth (validity) of the statement that Q data items satisfy the statement (summarizer) S.

First, we should comment on the form of the basic elements of the summary, i.e., the summarizer, quantity in agreement, and how to calculate the degree of truth (validity).

2.1. On the Form of the Summarizer

Since the only fully natural and human consistent means of communication for the humans is a natural language, we assume that the summarizer S is a linguistic expression semantically represented by a fuzzy set. For instance, in our example a summarizer like 'young' would be represented as a fuzzy set in the universe of discourse as, say, $\{1, 2, \ldots, 90\}$, i.e., containing possible values of the human age, and 'young' could be given as, say, a fuzzy set with a nonincreasing membership function in that universe such that, in a simple case of a piecewise linear membership function, the age up to 35 years is for sure 'young', i.e., the grade of membership is equal to 1, the age over 50 years is for sure 'not young', i.e., the grade of membership is equal to 0, and for the ages between 35 and 50 years the grades of membership are between 1 and 0, the higher the age the lower its corresponding grade of membership. Clearly, the meaning of the summarizer, i.e., its corresponding fuzzy set is in practice subjective, and may be either predefined or elicited from the user when needed.

Such a simple one-attribute-related summarizer exemplified by 'young' does well serve the purpose of introducing the concept of a linguistic summary, hence it was assumed by Yager (1982). However, it is of less practical relevance. It can be extended, for some confluence of attribute values as, e.g, '*young and well paid*', and then to more complicated combinations.

Clearly, when we try to linguistically summarize data, the most interesting are non-trivial, *human-consistent* summarizers (concepts) as, e.g.:

- *productive* workers,
- *stimulating* work environment,
- *difficult* orders, etc.

involving complicated *combinations of attributes*, e.g.: a hierarchy (not all attributes are of the same importance), the attribute values are ANDed and/or ORed, k out of n, most, etc. of them should be accounted for, etc.

The generation and processing of such non-trivial summarizers needs some specific tools and techniques to be discussed later.

2.2. On the Form of the Quantity in Agreement

The quantity in agreement, Q, is a proposed indication of the extent to which the data satisfy the summary. Once again, a precise indication is not human consistent, and a linguistic term represented by a fuzzy set is employed.

Basically, two types of such a linguistic quantity in agreement can be used:

- absolute as, e.g., 'about 5', 'more or less 100', 'several', and
- relative as, e.g., 'a few', 'more or less a half', 'most', almost all, etc.

Notice that the above linguistic expressions are the so-called fuzzy linguistic quantifiers (cf. Zadeh, 1983; 1985) that can be handled by fuzzy logic.

As for the fuzzy summarizer, also in the case of a fuzzy quantity in agreement, its form is subjective, and can be either predefined or elicited from the user when needed.

2.3. Calculation of the Truth (Validity) of a Linguistic Summary

Basically, the calculation of the truth (validity) of the basic type of a linguistic summary considered in this section is equivalent to the calculation of the truth value (from the unit interval) of a linguistically quantified statement (e.g., '*most* of the employees are *young*'). This may be done by two most relevant techniques using either Zadeh's (1983) calculus of linguistically quantified statements (cf. Zadeh and Kacprzyk, 1992) or Yager's (1988) OWA operators (cf. Yager and Kacprzyk, 1997); for a survey, see Liu and Kerre (1998).

A linguistically quantified proposition, exemplified by 'most experts are convinced', is written as 'Qy's are F' where Q is a linguistic quantifier (e.g., most), $Y = \{y\}$ is a set of objects (e.g., experts), and F is a property (e.g., convinced). Importance B may be added yielding 'QBy's are F', e.g., 'most (Q) of the important (B) experts (y's) are convinced (F)'. The problem is to find truth(Qy's are F) or truth(QBy's are F), respectively, knowing truth(y is F), $\forall y \in Y$ which is done here using Zadeh's (1983) fuzzy-logic-based calculus of linguistically quantified propositions.

Property F and importance B are fuzzy sets in Y, and a (proportional, nondecreasing) linguistic quantifier Q is assumed to be a fuzzy set in $[0, 1]$ as, e.g.

$$\mu_Q(x) = \begin{cases} 1 & \text{for } x \geq 0.8, \\ 2x - 0.6 & \text{for } 0.3 < x < 0.8, \\ 0 & \text{for } x \leq 0.3. \end{cases} \tag{1}$$

Then, due to Zadeh (1983)

$$\text{truth}\,(Qy\text{'s are }F) = \mu_Q\left[\frac{1}{n}\sum_{i=1}^{n}\mu_F(y_i)\right], \tag{2}$$

$$\text{truth}\,(QBy\text{'s are }F) = \mu_Q\left[\frac{\sum\limits_{i=1}^{n}\left(\mu_B(y_i)\wedge\mu_F(y_i)\right)}{\sum\limits_{i=1}^{n}\mu_B(y_i)}\right]. \tag{3}$$

An OWA operator (Yager, 1988; Yager and Kacprzyk, 1997) of dimension p is a mapping $F:[0,1]^p \to [0,1]$ if associated with F is a weighting vector $W = [w_1,\ldots,w_p]^T$, $w_i \in [0,1], w_1+\cdots+w_p = 1$, and

$$F(x_1,\ldots,x_p) = w_1 b_1 + \cdots + w_p b_p = W^T B, \tag{4}$$

where b_i is the i-th largest element among $x_1,\ldots,x_p$, $B = [b_1,\ldots,b_p]$.

The OWA weights can be found from the membership function of Q due to (cf. Yager, 1988):

$$w_i = \begin{cases} \mu_Q(i) - \mu_Q(i-1) & \text{for } i = 1,\ldots,p, \\ \mu_Q(0) & \text{for } i = 0. \end{cases} \tag{5}$$

The OWA operators can model a wide array of aggregation operators (including linguistic quantifiers), from $w_1 = \cdots = w_{p-1} = 0$ and $w_p = 1$ which corresponds to 'all', to $w_1 = 1$ and $w_2 = \cdots = w_p = 0$ which corresponds to 'at least one', through all intermediate situations.

An important issue is related to the OWA operators for importance qualified data. Suppose that we have $A = [a_1,\ldots,a_p]$, and a vector of importances $V = [v_1,\ldots,v_p]$ such that $v_i \in [0,1]$ is the importance of a_i, $i = 1,\ldots,p$, $v_1+\cdots+v_p = 1$. The case of an *ordered weighted averaging operator with importance qualification*, denoted by OWA_Q, is not trivial. In Yager's (1988) approach to be used here, which seems to be highly plausible (though is sometimes criticized), some redefinition of the OWA's weights w_i's into $\bar{w}_i$'s is performed, and (4) becomes

$$F_I(x_1,\ldots,x_p) = \bar{w}_1 b_1 + \cdots \bar{w}_p b_p = \bar{W}^T B, \tag{6}$$

where

$$w_j = \mu_Q\left(\frac{\sum\limits_{k=1}^{j} u_k}{\sum\limits_{k=1}^{p} u_k}\right) - \mu_Q\left(\frac{\sum\limits_{k=1}^{j-1} u_k}{\sum\limits_{k=1}^{p} u_k}\right), \tag{7}$$

where u_k is the importance of b_k, i.e., the k-largest element of A (i.e., the corresponding v_k).

3. Some Other Validity Criteria

The basic validity criterion, i.e., the truth of a linguistically quantified statement given by (2) and (3), is certainly the most important in the general framework assumed. However, it does not grasp all the aspects of a linguistic summary. Some attempts at devising other quality (validity) criteria will be briefly surveyed following Kacprzyk and Yager (2000).

First, Yager (1982; 1991) proposed a measure of informativeness that may be summarized as follows. Suppose that we have a data set whose elements are from a measurement space X. One can say that the data set itself is its own most informative description, and any other summary implies a loss of information. So, a natural question is whether a particular summary is informative, and to what extent.

It turns out that the degree of truth used so far is not a good measure of informativeness (cf. Yager, 1982; 1991). Let the summary be characterized by the triple $(S,\ Q,\ T)$, and let a related summary be characterized by the triple $(S^c,\ Q^c,\ T)$, such that S^c is the negation of S, i.e., $\mu_S^c(\cdot) = 1 - \mu_S(\cdot)$ and $\mu_Q^c(\cdot) = 1 - \mu_Q(\cdot)$.

Then Yager (1982; 1991) proposes the following measure of informativeness of a summary:

$$I = \left[T\ SP(Q)\ SP(S)\right] \vee \left[(1-T)Sp(Q^c)Sp(S^c)\right], \tag{8}$$

where $SP(Q)$ is the specificity of Q given as

$$SP(Q) = \int_0^1 \frac{1}{\operatorname{card} Q_\alpha} \mathrm{d}\alpha, \tag{9}$$

where Q_α denotes the α-cut of Q, card$(\cdot)$ is the cardinality of the respective set, and similarly for Q^c, S, S^c.

This measure of informativeness results from a very plausible reasoning which can be found, e.g., in Yager (1982; 1991).

Unfortunately, though the above measure of informativeness is plausible and constitutes a considerable step forward, it is by no means a definite solution. First, let us briefly mention George and Srikanth's (1996) proposal. Suppose that a linguistic summary of interest involves more than 1 attribute (e.g., 'age', 'salary' and 'seniority' in the case of employees). Basically, for the same set of data, two summaries are generated:

- a *constraint descriptor* which is the most specific description (summary) that fits the largest number of tuples in the relation (database) involving the attributes in question,

- a *constituent descriptor* which is the description (summary) that fits the largest subset of tuples with the condition that each tuple attribute value takes on at least a threshold value of membership.

George and Srikanth (1996) use these two summaries to derive a fitness function (goodness of a summary) that is later used for deriving a solution (a best summary) via a genetic algorithm they employ. This fitness function represents a compromise between the most specific summary (corresponding to the constraint descriptor) and the most general summary (corresponding to the constituent descriptor).

Then, Kacprzyk (1999a; 1999b), and Kacprzyk and Strykowski (1999a; 1999b) propose some additional measures that have been further developed by Kacprzyk and Yager (2000).

For convenience of the reader, let us briefly repeat some basic notation. We have a data set (database) D that concerns some objects (e.g., employees) $Y = \{y_1, \dots, y_n\}$ described by some attribute V (e.g., age) taking on values in a set $X = \{x_1, x_2, \dots\}$ exemplified by $\{20, 21, \dots, 100\}$ or {very young, young, ..., old, very old}. Let $d_i = V(y_i)$ denote the value of attribute V for object y_i. Therefore, the data set to be summarized is given as a table

$$D = [d_1, \dots, d_n] = \big[V(y_1), V(y_2), \dots, V(y_n)\big]. \tag{10}$$

In a more realistic case the data set is described by more than one attribute, and let $V = \{V_1, V_2, \dots, V_m\}$ be a set of such attributes taking values in X_i, $i = 1, \dots, m$; $V_j(y_i)$ denotes the value of attribute V_j for object y_i, and attribute V_j takes on its values from a set X_j.

The data set to be summarized is therefore:

$$\begin{aligned} D = \Big\{ & \big[V_1(y_1), V_2(y_1), \dots, V_m(y_1)\big], \big[V_1(y_2), V_2(y_2), \dots, V_m(y_2)\big], \\ & \dots, \big[V_1(y_n), V_2(y_n), \dots, V_m(y_n)\big] \Big\}. \end{aligned} \tag{11}$$

In this case of multiple attributes the description (summarizer) S is a family of fuzzy sets $S = \{S_1, S_2, \dots S_m\}$ where S_i is a fuzzy set in X_i, $i = 1, \dots, m$. Then, $\mu_S(y_i)$ may be defined as

$$\mu_S(y_i) = \min_{j \in \{1,2,\dots,m\}} \big[\mu_{S_j}\big(V_j(y_i)\big)\big]. \tag{12}$$

So, having S, we can calculate the truth value T of a summary for any quantity in agreement. However, to find a best summary, we should calculate T for each possible summarizer, and for each record in the database in question. This computational task is prohibitive for virtually all non-trivial databases and numbers of attributes.

A natural line of reasoning would be to either limit the number of attributes of interest or to limit the class of possible summaries by setting a more specific description by predefining a 'narrower' description (e.g., very young, young and well paid, etc. employees). This will limit the search space.

We will deal here with the second option. The user can limit the scope of a linguistic summary to, say, those for which the attribute 'age' takes on the value 'young (employees)' only, i.e., to fix the summarizer related to that attribute. That is, this will correspond to the searching of the database using the query w_g equated with the fuzzy set in X_g corresponding to 'young' related to attribute V_g (i.e., age), i.e., characterized by $\mu_{w_g}(\cdot)$. In such a case, $\mu_S(y_i)$ given by (12) becomes

$$\mu_S(y_i) = \min_{j \in \{1,2,\dots,m\}} \Big[\mu_{S_j}\big(V_j(y_i)\big) \wedge \mu_{w_g}\big(V_g(y_i)\big)\Big], \quad i = 1, \dots, n, \tag{13}$$

where '$\wedge$' is the minimum (or, more generally, a t-norm), and then

$$r = \frac{\sum_{i=1}^{n} \mu_S(y_i)}{\sum_{i=1}^{n} \mu_{w_g}(V_g(y_i))} \tag{14}$$

and $T = \mu_Q(r)$.

Now, we will present the five new quality measures of linguistic database summaries introduced in (Kacprzyk, 1999a; 1999b; Kacprzyk and Strykowski, 1999a; 1999b; Kacprzyk and Yager, 2000):

- a truth value which basically corresponds to the degree of truth of a lingustically quantified proposition representing the summary given by, say, (2) or (3),
- a degree of imprecision (fuzziness),
- a degree of covering,
- a degree of appropriateness,
- a length of a summary,

and these degrees will now be formally defined.

For notational simplicity later on, let us rewrite (13) and (14) as

$$\mu_S(d_i) = \min_{j \in \{1,2,\dots,m\}} \left[(\mu_{S_j}(V_j(y_i)) \right], \quad i = 1, \dots, n \tag{15}$$

and

$$r = \frac{\sum_{i=1}^{n} [\mu_S(y_i) \wedge \mu_{w_g}(V_g(y_i))]}{\sum_{i=1}^{n} \mu_{w_g}(V_g(y_i))}. \tag{16}$$

The *degree of truth*, T_1, is the basic validity criterion introduced in (Yager's, 1982; 1991) and commonly employed. It is clearly equal to

$$T_1 = \mu_Q(r) \tag{17}$$

and (17) results clearly from Zadeh's (1983) calculus of linguistically quantified propositions.

The *degree of imprecision (fuzziness, specificity)* is an obvious and important validity criterion. Basically, a very imprecise (fuzzy) linguistic summary (e.g., on almost all winter days the temperature is rather cold) has a very high degree of truth yet it is not useful.

Suppose that description (summarizer) S is given as a family of fuzzy sets $S = \{s_1, s_2, \dots s_m\}$. For a fuzzy set s_j, $j = 1, \dots, m$, we can define its degree of fuzziness as, e.g.:

$$\mathrm{in}(s_j) = \frac{\mathrm{card}\,\{x \in X_j : \mu_{s_j}(x) > 0\}}{\mathrm{card}\,X_j}, \tag{18}$$

where card denotes the cardinality of the corresponding (nonfuzzy) set. That is, the 'flatter' the fuzzy set s_j, the higher the value of $\mathrm{in}(s_j)$. The degree of imprecision (fuzziness), T_2, of the summary—or, in fact, of S—is then defined as

$$T_2 = 1 - \sqrt[m]{\prod_{j=1,\dots,m} \mathrm{in}\,(s_j)}. \tag{19}$$

Notice that the degree of imprecision T_2 depends on the form of the summary only and not on the database, that is its calculation does not require the searching of the database (all its records) which is very important.

The *degree of covering*, T_3, is defined as

$$T_3 = \frac{\sum_{i=1}^{n} t_i}{\sum_{i=1}^{n} h_i}, \tag{20}$$

where

$$t_i = \begin{cases} 1 & \text{if} \quad \mu_S(y_i) > 0 \quad \text{and} \quad \mu_{w_g}\big(V_g(y_i)\big) > 0, \\ 0 & \text{otherwise,} \end{cases}$$

$$h_i = \begin{cases} 1 & \text{if} \quad \mu_{w_g}\big(V_g(y_i)\big) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

The degree of covering says how many objects in the data set (database) corresponding to the query w_g are 'covered' by the particular summary, i.e. to the particular description S. Its interpretation is simple as, e.g., if it is equal to 0.15, then this means that 15% of the objects are consistent with the summary in question. The value of this degree depends clearly on the contents of the database.

The *degree of appropriateness* is the most relevant degree of validity. To present its idea, suppose that the summary containing the description (fuzzy sets) $S = (S_1, S_2, \dots, S_m)$ is partitioned into m partial summaries each of which encompasses the particular attributes $S_1, S_2, \dots, S_m$, such that each partial summary corresponds to one fuzzy value only. Then if we set

$$S_j(y_i) = \mu_{S_j}\big(V_j(y_i)\big), \tag{21}$$

then

$$r_j = \frac{1}{n}\sum_{i=1}^{n} h_i, \quad j = 1, \ldots, n, \tag{22}$$

where

$$h_i = \begin{cases} 1 & \text{if} \quad S_j(y_i) > 0, \\ 0 & \text{otherwise,} \end{cases}$$

and the degree of appropriateness, T_4, is defined as

$$T_4 = \operatorname{abs}\Big(\prod_{j=1,\ldots,m} r_j - T_3 \Big). \tag{23}$$

The degree of appropriateness means that, for a database concerning the employees, if 50% of them are less than 25 year old and 50% are highly qualified, then we may expect that 25% of the employees would be less than 25 year old and highly qualified; this would correspond to a typical, fully expected situation. However, if the degree of appropriateness is, say, 0.39 (i.e., 39% are less than 25 years old and highly qualified), then the summary found reflects an interesting, not fully expected relation in our data. This degree describes therefore how characteristic for the particular database the summary found is. T_4 is very important because, for instance, a trivial summary like '100% of sale is of any articles' has full validity (truth) if we use the traditional degree of truth but its degree of appropriateness is equal 0 which is correct.

The *length* of a summary is relevant because a long summary is not easily comprehensible by the human user. This length, T_5, may be defined in various ways, and the form below has proven to be useful:

$$T_5 = 2\big(0.5^{\operatorname{card} S}\big), \tag{24}$$

where $\operatorname{card} S$ is the number of elements in S.

Now, the (total) degree of validity, T, of a particular linguistic summary is defined as the weighted average of the above five degrees of validity, i.e.:

$$T = T(T_1, T_2, T_3, T_4, T_5;\ w_1, w_2, w_3, w_4, w_5) = \sum_{i=1,2,\ldots,5} w_i T_i \tag{25}$$

and the problem is to find an optimal summary, $S^* \in \{S\}$, such that

$$S^* = \arg\max_S \sum_{i=1,2,\ldots,5} w_i T_i, \tag{26}$$

where $w_1, \ldots, w_5$ are weights assigned to the particular degrees of validity, with values from the unit interval, the higher being the more important such that $\sum_{i=1,2,\ldots,5} w_i = 1$.

The definition of weights, $w_1, \ldots, w_5$, is a problem in itself, and will not be dealt with in more detail. The weights can be predefined or elicited from the user. In the case study presented later on the weights are determined by using the well-known Saaty's (1980) AHP (analytical hierarchy process) approach that works well in the problem considered.

4. Derivation of Linguistic Summaries via a Fuzzy Logic Based Querying Interface

The roots of the approach adopted are our previous papers on the use of fuzzy logic in querying databases (cf. Kacprzyk and Zadrożny, 1994; 1995a; 1995b; 1996; 1997; Kacprzyk *et al.*, 1989; Zadrożny and Kacprzyk, 1995) in which we argued that the formulation of a precise query is often difficult for the end user (see also Zemankova and Kacprzyk, 1993). For example, a customer of a real-estate agency looking for a house would rather express his or her criteria using imprecise descriptions as a *cheap*, *large* garden, etc. Also, to specify which combination of the criteria fulfilment would be satisfactory, he or she would often use, say, *most* of them or *almost* all. All such vague terms may be relatively easily interpreted using fuzzy logic. This has motivated the development of the whole family of fuzzy querying interfaces, notably our FQUERY for the Access package (cf. Kacprzyk and Zadrożny, 1994; 1995a; 1995b; 1996; 1997; Zadrożny and Kacprzyk, 1995).

The same arguments apply, to an even higher degree, when one tries to summarize the content of a database in a short (linguistic) statement. For example, a summary like '*most* our customers are *reliable*' may be more useful than, say '65% of our customers have paid at least 70% of their duties in less than 10 days'.

In the previous section we studied the summarization independently, and here we will restate it in the fuzzy querying context. We start with the reinterpretation of (2) and (3). Thus, (2) formally expresses a statement:

$$\text{'}Most\ records\ match\ query\ S\text{'} \tag{27}$$

where S replaces F in (2) since we refer here directly to the concept of a summarizer.

We assume a standard meaning of the query as a set of conditions on the values of fields from the database's tables, connected with AND and OR. We allow for fuzzy terms in a query which implies a degree of matching from [0,1] rather than a yes/no matching. So, a query S defines a fuzzy subset (fuzzy property) on the set of the records, and the membership of them is determined by their matching degree with the query.

Similarly, (3) may be interpreted as expressing a statement of the following type:

$$\text{'}Most\ records\ meeting\ conditions\ F\ match\ query\ S\text{'} \tag{28}$$

Thus, (28) says something only about a subset of records taken into account by (27). That is, in the database terminology, F corresponds to a *filter* and (28) claims that most records passing through F match query S. Moreover, since the filter may be fuzzy, a record may pass through it to a degree from [0,1].

We seek, for a given database, propositions of the type (3), interpreted as (28) that are highly true, and they consist of three elements: a fuzzy filter F (optional), a query S, and a linguistic quantifier Q. There are two limit cases, where we:

- do not assume anything about the form of any of these elements,
- assume concrete forms of the fuzzy filter and the query, and look only for a linguistic quantifier Q.

In the first case data summarization will be extremely time-consuming, but may produce interesting results. In the second case the user has to guess a good candidate formula for summarization but the evaluation is fairly simple being equivalent to the answering of a (fuzzy) query. Thus, the second case refers to the summarization known as *ad hoc queries*, extended with an automatic determination of a linguistic quantifier.

In between these two extreme cases, there are different types of summaries, with various assumptions on what is given and what is sought. In the case of a linguistic quantifier the situation is simple: it may be given or sought. In the case of a fuzzy filter F and a fuzzy query S, more possibilities exist as both F and S consist of simple conditions, each stating what *value* a *field* should take on, and connected using logical connectives. Here we assume that the table(s) of interest for summarization are fixed.

We will use the following notation to describe what is given or what is sought with respect to the fuzzy filter F and query S (below A stands for either F or S):

- A — all is given (or sought), i.e., attributes, values and the structure,
- A^{fc} — attributes and the structure are given, but the values are left out,
- A^{v} — denotes sought left out values referred to in the above notation, and
- A^{f} — only a set of attributes is given, the other elements are sought.

Using the above notation, we can propose the classification of the summaries shown in Table 1 (see also Kacprzyk and Zadrożny, 2000a; 2000b; 2000c), and explained in more detail below.

Table 1. Classification of linguistic summaries.

Type	Given	Sought	Remarks
1	S	Q – sought	Simple summaries through ad-hoc queries
2	$S\,F$	Q – sought	Conditional summaries through ad-hoc queries
3	$Q\,S^{fc}$	S^{v}	Simple value oriented summaries
4	$Q\,S^{fc}\,F$	S^{v}	Conditional value oriented summaries
5	nothing	$S\,F\,Q$	General fuzzy rules

Thus, we distinguish 5 main types of data summaries. Type 1 can be easily produced by a simple extension of fuzzy querying as proposed and implemented in our FQUERY for Access package, see Section 6 for more details. Basically, the user has to construct a query—a candidate summary. Then, it has to be determined what is the fraction of the rows matching this query and what linguistic quantifier best denotes this fraction. The primary target of this type of summarization is certainly to propose such a query that a large proportion, e.g., *most*, of the rows satisfies it. On the other hand, it may be interesting to learn that only *few* rows satisfy some meaningful query. A Type 2 summary is a straight extension of Type 1 summaries by adding a fuzzy filter. As soon as a fuzzy querying engine deals with fuzzy filters, the computational complexity of this type of summaries is the same as for Type 1. For more on these types of summaries, see for instance (Anwar *et al.*, 1992; Kacprzyk and Zadrożny, 1998).

The summaries of Type 3 require much more effort. A primary goal of this type of summary is to determine typical (exceptional) values of an attribute. In such a special case, query S consists of only one simple condition built of the attribute whose typical (exceptional) value is sought, the '=' relational operator and a placeholder for the value sought. For example, using the following summary in a context of personal data:

$$Q = \text{'}most\text{'} \text{ and } S = \text{'age=?'}$$

('?' denotes the placeholder mentioned above) we look for a typical value of the age of the employees.

Then, we try to find a (possibly fuzzy) value such that the query matches to a high degree Q of the rows. Depending on the category of the Q used as, e.g., *most* versus *few*, typical or exceptional values are sought, respectively. Some more considerations are required as in some cases all values may turn out to be exceptional and none to be typical. This type of summaries may be used with more complicated, regular queries but it may quickly become computationally infeasible (due to the combinatorial explosion) and the interpretation of results becomes vague. A Type 4 summary may produce typical (exceptional) values for some, possibly fuzzy, subset of rows. From the computational point of view, the same remarks apply as for Type 1 versus Type 2 summaries.

A Type 5 summary represents the most general form considered here. In its full version this type of summaries is to produce fuzzy rules describing dependencies between specific values of particular attributes. Here the use of the filter is essential, in contrast to the previous types where it was optional. The very meaning of a fuzzy rule obtained is that if a row meets a filter's condition, then it meets also the query's conditions—this corresponds to a classical IF-THEN rule. For a general form of such a rule it is difficult to devise an effective and efficient generation algorithm. A full search may be acceptable only in the case of restrictively limited sets of rule building blocks, i.e., attributes and their possible values. Here, some genetic algorithm based approaches may be employed (cf. George and Srikanth, 1996) to alleviate the computational complexity, additional assumptions may also be made. For example, some sets of relevant (interesting, promising, etc.) attributes for the query (S^f) and the filter (F^f) may be selected in advance. Some constraints may also be put on the structure of the query S and filter F (in terms of the number of logical connectives allowed). Another important special case of Type 5 summaries refers to the situation where the query (S) is fixed and only the filter (F) and quantifier (Q) are sought, i.e., we look for causes of given data features. For example, we may set in a query that profitability of a venture is *high* and look for the characterization of ventures (rows) securing such a high profitability.

The summaries of Types 1 and 3 have been implemented as an extension to our FQUERY for Access.

FQUERY for Access is an add-in that makes it possible to use fuzzy terms in queries (Kacprzyk and Zadrożny, 1994; 1995a; 1995b; 1996; 1997; Zadrożny and Kacprzyk, 1995). Briefly speaking, the following types of fuzzy terms are available:

- fuzzy values, exemplified by *low* in 'profitability is *low*',
- fuzzy relations, exemplified by *much greater than* in 'income is *much greater than* spending', and
- linguistic quantifiers, exemplified by *most* in '*most* conditions have to be met'.

The elements of the first two types are elementary building blocks of fuzzy queries in FQUERY for Access. They are meaningful in the context of numerical fields only. There are also other fuzzy constructs allowed which may be used with scalar fields.

If a field is to be used in a query in connection with a fuzzy value, it has to be defined as an *attribute*. The definition of an attribute consists of two numbers: the attribute's values lower (LL) and upper (UL) limit. They set the interval which the field's values are assumed to belong to, according to the user. This interval depends on the meaning of the given field. For example, for *age* (of a person), the reasonable interval would be, e.g., [18,65], in a particular context, i.e., for a specific group. Such a concept of an attribute makes it possible to universally define fuzzy values.

Fuzzy values are defined as fuzzy sets on $[-10, +10]$. Then, *the matching degree* $\mathrm{md}(\cdot,\cdot)$ of a simple condition referring to attribute AT and fuzzy value FV against a record R is calculated by:

$$\mathrm{md}(\mathrm{AT} = \mathrm{FV}, \mathrm{R}) = \mu_{\mathrm{FV}}\left(\tau(\mathrm{R}(\mathrm{AT}))\right),$$

where R(AT) is the value of attribute AT in record R, μ_{FV} is the membership function of fuzzy value FV, $\tau : [\mathrm{LL}_{\mathrm{AT}}, \mathrm{UL}_{\mathrm{AT}}] \to [-10, 10]$ is the mapping from the interval defining AT onto $[-10, 10]$ so that we may use the same fuzzy values for different fields. A meaningful interpretation is secured by τ which makes it possible to treat all fields domains as ranging over the unified interval $[-10, 10]$.

For simplicity, it is assumed that the membership functions of fuzzy values are trapezoidal as in Fig. 1 and τ is assumed linear.

Linguistic quantifiers allow for a flexible aggregation of simple conditions. In FQUERY for Access the fuzzy linguistic quantifiers are defined in Zadeh's (1983) sense (see Section 2), as a fuzzy set on $[0, 10]$ interval instead of the original $[0, 1]$. They may be interpreted either using Zadeh's original approach or via the OWA operators (cf. Yager, 1988; Yager and Kacprzyk, 1997); Zadeh's interpretation will be used here. The membership functions of fuzzy linguistic quantifiers are assumed

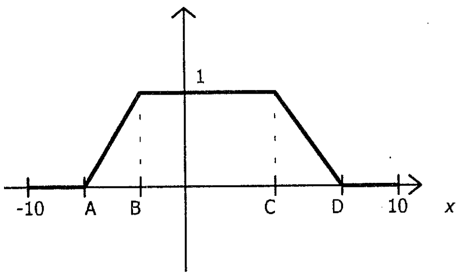

Fig. 1. An example of the membership function of a fuzzy value.

piece-wise linear, hence two numbers from $[0,10]$ are needed. Again, a mapping from $[0,N]$, where N is the number of conditions aggregated, to $[0,10]$ is employed to calculate the matching degree of a query. More precisely, the matching degree, $\mathrm{md}(\cdot,\cdot)$, for the *query* 'Q of N conditions are satisfied' for record R is equal to

$$\mathrm{md}(Q,\ \mathrm{condition_i},\ \mathrm{R}) = \mu_Q\left[\tau\Big(\sum_i \mathrm{md}(\mathrm{condition_i}, \mathrm{R})\Big)\right].$$

We can also assign different importance degrees for particular conditions. Then, the aggregation formula is equivalent to (3). The importance is identified with a fuzzy set on $[0,1]$, and then treated as property B in (3).

FQUERY for Access has been designed so that queries containing fuzzy terms are still syntactically correct Access's queries. It has been attained through the use of parameters. Basically, Access represents the queries using SQL. Parameters, expressed as strings limited with brackets, make it possible to embed references to fuzzy terms in a query. We have assumed a special naming convention for the parameters corresponding to particular fuzzy terms. For example, a parameter like:

[FfA_FV fuzzy value name]	will be interpreted as a fuzzy value,
[FfA_FQ fuzzy quantifier name]	will be interpreted as a fuzzy quantifier.

Before a fuzzy term may be used in a query, it has to be defined using the toolbar provided by FQUERY for Access and stored internally. This feature, i.e., maintenance of dictionaries of fuzzy terms defined by users, strongly supports our approach to data summarization to be discussed next. In fact, the package comes with a set of predefined fuzzy terms but the user may enrich the dictionary too.

When the user initiates the execution of a query, it is automatically transformed by appropriate routines of FQUERY for Access and then run as a native query of Access. The transformation consists primarily in the replacement of parameters referring to fuzzy terms by calls to functions implemented by the package which secure a proper interpretation of these fuzzy terms. Then, the query is run by Access as usual.

FQUERY for Access provides its own toolbar. There is one button for each fuzzy element, and the buttons for declaring attributes, starting the querying, closing the toolbar and for help (cf. Fig. 2). Details can be found in (Kacprzyk and Zadrożny, 1994; 1995a; 1995b; 1996; 1997; Zadrożny and Kacprzyk, 1995).

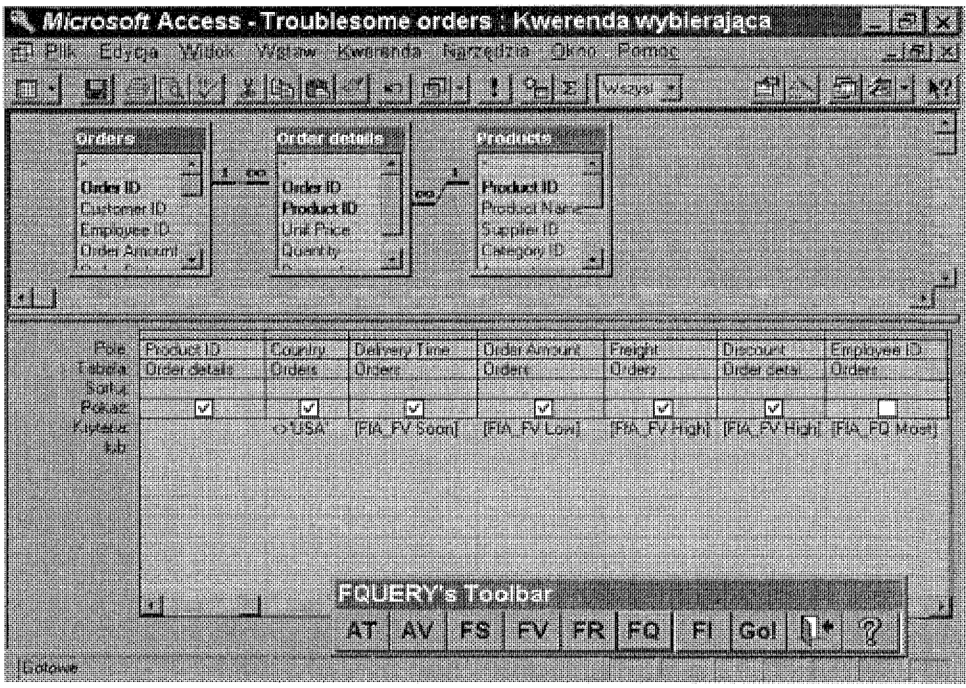

Fig. 2. Composition of a fuzzy query.

5. An Example of Implementation

The proposed data summarization procedure was implemented on a sales database of a computer retailer in Southern Poland (cf. Kacprzyk, 1999a; 1999b; Kacprzyk and Strykowski, 1999a; 1999b). The basic structure of the database is as shown in Table 2.

First, after some initialization, we provide the parameters concerning mainly: the definition of attributes and the subject, the definition of how the results should be presented, and the definition of parameters of the method (i.e. a genetic algorithm or, seldom, a full search). Then, we initialize the search, and obtain the results shown in the next tables in which the consecutive columns contain: the summary, values of the 4 indicators, i.e. the degrees of appropriateness, covering, truth, and fuzziness (the length is not accounted for in our simple case), and finally the weighted average (with weights chosen after some learning and a fine tuning phase).

We will now give a couple of examples. First, if we are interested in the relation between the commission and the type of goods sold, then we obtain the linguistic summaries shown in Table 3.

Table 2. Structure of the database.

Attribute name	Attribute type	Description
Date	Date	Date of sale
Time	Time	Time of sale transaction
Name	Text	Name of the product
Amount (number)	Numeric	Number of products sold in the transaction
Price	Numeric	Unit price
Commission	Numeric	Commission (in %) on sale
Value	Numeric	Value = amount (number) $\times$ price of the product
Discount	Numeric	Discount (in %) for transaction
Group	Text	Product group to which the product belongs
Transaction value	Numeric	Value of the whole transaction
Total sale to customer	Numeric	Total value of sales to the customer in fiscal year
Purchasing frequency	Numeric	Number of purchases by customer in fiscal year
Town	Text	Town where the customer lives

Table 3. Linguistic summaries expressing relations between the group of products and commission.

Summary	Degree of appropriateness	Degree of covering	Degree of validity
	Degree of imprecision	Weighted average	
About 1/2 of sales of network elements is with a high commission	0.2329 0.1872	0.4202 0.3165	0.3630
About 1/2 of sales of computers is with a medium commission	0.2045 0.3453	0.5498 0.3699	0.4753
Much sales of accessories is with a high commission.	0.1684 0.4095	0.5779 0.3919	0.5713
Much sales of components is with a low commission	0.1376 0.5837	0.7212 0.4449	0.6707
About 1/2 of sales of software is with a low commission	0.1028 0.5837	0.4808 0.3162	0.4309
About 1/2 of sales of computers is with a low commission	0.0225 0.5837	0.5594 0.3202	0.4473
A few sales of components is without commission	0.0237 0.2745	0.0355 0.2346	0.0355
A few sales of computers is with a high commission	0.1418 0.1872	0.0455 0.1881	0.0314
Very few sales of printers is with a high commission	0.1288 0.1872	0.0585 0.1820	0.0509

As we can see, the results can be very helpful in, e.g., negotiating commissions for various products sold.

Next, suppose that we are interested in relations between the groups of products and times of sale. We obtain the results as in Table 4. Notice that in this case the summaries are much less obvious than in the former case.

Table 4. Linguistic summaries expressing relations between the groups of products and times of sale.

Summary	Degree of appropriateness Degree of imprecision	Degree of covering Weighted average	Degree of validity
About 1/3 of sales of computers is by the end of year	0.0999 0.2010	0.3009 0.1274	0.2801
About 1/2 of sales in autumn is of accessories	0.0642 0.4095	0.4737 0.1143	0.4790
About 1/3 of sales of network elements is in the beginning of year	0.0733 0.2124	0.2857 0.0982	0.1957
Very few sales of network elements is by the end of year	0.0833 0.2010	0.1176 0.0980	0.0929
Very few sales of software is in the beginning of year	0.0768 0.2124	0.1355 0.0929	0.0958
About 1/2 of sales in the beginning of year is of accessories	0.0348 0.4095	0.4443 0.0860	0.4343
About 1/3 of sales in the summer is of accessories	0.0464 0.2745	0.3209 0.0853	0.3092
About 1/3 of sales of peripherals is in the spring period	0.0507 0.2525	0.3032 0.0809	0.2140
About 1/3 of sales of software is by the end of year	0.0446 0.2010	0.2455 0.0768	0.2258
About 1/3 of sales of network elements is in the spring period	0.0458 0.2525	0.2983 0.0763	0.2081
About 1/3 of sales in the summer period is of components	0.0336 0.2745	0.3081 0.0745	0.3081
Very few sales of network elements is in the autumn period	0.0485 0.1956	0.1471 0.0692	0.0955
A few sales of software is in the summer period	0.0402 0.1362	0.1765 0.0691	0.1765

Finally, let us show in Table 5 some of the obtained linguistic summaries expressing relations between the attributes: size of customer, regularity of customer (purchasing frequency), date of sale, time of sale, commission, group of product and day of sale. This is an example of the most sophisticated form of linguistic summaries allowed in the system. These sets of most valid summaries (normally, not just one summary) will give much insight into relations between the attributes chosen.

Table 5. Linguistic summaries expressing relations between the attributes: size of customer, regularity of customer (purchasing frequency), date of sale, time of sale, commission, group of product and day of sale.

Summary	Degree of appropriateness / Degree of imprecision	Degree of covering / Weighted average	Degree of validity
Much sales on Saturday is about noon with a low commission	0.3843 0.2748	0.6591 0.3863	0.3951
Much sales on Saturday is about noon for bigger customers	0.3425 0.4075	0.7500 0.3648	0.4430
Much sales on Saturday is about noon	0.3133 0.4708	0.7841 0.3564	0.4654
Much sales on Saturday is about noon for regular customers	0.3391 0.3540	0.6932 0.3558	0.4153
A few sales for regular customers is with a low commission	0.3882 0.5837	0.1954 0.3451	0.1578
A few sales for small customers is with a low commission	0.3574 0.5837	0.2263 0.3263	0.1915
A few sales for one-time customers is with a low commission	0.3497 0.5837	0.2339 0.3195	0.1726
Much sales for small customers is for nonregular customers	0.6250 0.1458	0.7709 0.5986	0.5105

6. Concluding Remarks

We have presented two extensions of Yager's (1982; 1989; 1991; 1994; 1995a; 1995b; 1996) general approach to the linguistic summarization of a set of data (database). The first extension is mainly through the use of additional degrees of validity (quality) namely those of: truth, imprecision (fuzziness), covering, appropriateness, and length whose weighted average is the quality (performance) measure of a linguistic summary. The other approach involves an extension of the structure of a linguistic summary, and embedding the summarization procedure within a flexible (fuzzy) querying environment. Finally, we present an application of the first approach for the derivation of linguistic summaries of a sales database at a computer retailer, and show that the summaries obtained may be of a considerable practical value for management.

References

Anwar T.M., Beck H.W. and Navathe S.B. (1992): *Knowledge mining by imprecise querying: A classification based system.* — Proc. Int. Conf. *Data Engineering*, Tampa, USA, pp.622–630.

George R. and Srikanth R. (1996): *Data summarization using genetic algorithms and fuzzy logic*, In: Genetic Algorithms and Soft Computing (F. Herrera and J.L. Verdegay, Eds.). — Heidelberg: Physica-Verlag, pp.599–611.

Kacprzyk J. (1999a): *An interactive fuzzy logic approach to lingustic data summaries.* — Proc. 18th Int. Conf. *North American Fuzzy Information Processing Society, NAFIPS'99*, IEEE, pp.595–599.

Kacprzyk J. (1999b): *A new paradigm shift from computation on numbers to computation on words on an example of linguistic database summarization*, In: Emerging Knowledge Engineering and Connectionist-Based Information Systems (N. Kasabov, Ed.). — Proc. ICONIP/ANZIIS/ANNES'99, University of Otago, pp.179–179.

Kacprzyk J. and Strykowski P. (1999a): *Linguistic data summaries for intelligent decision support*, In: Fuzzy Decision Analysis and Recognition Technology for Management, Planning and Optimization (R. Felix, Ed.). — Proc. EFDAN'99, FLS Systeme G.m.B.H., pp.3–12.

Kacprzyk J. and Strykowski P. (1999b): *Linguitic summaries of sales data at a computer retailer: A case study.* — Proc. IFSA'99, Taipei, Taiwan R.O.C, Vol.1, pp.29–33.

Kacprzyk J. and Yager R.R. (2000): *Linguistic summaries of data using fuzzy logic.* — Int. J. General Syst. (in press).

Kacprzyk J., Zadrożny S. and Ziółkowski A. (1989): *FQUERY III+: a 'human consistent' database querying system based on fuzzy logic with linguistic quantifiers.* — Inform. Syst., Vol.6, pp.443–453.

Kacprzyk J. and Zadrożny S. (1994): *Fuzzy querying for Microsoft Access.* — Proc. FUZZ-IEEE'94, Orlando, USA, Vol.1, pp.167–171.

Kacprzyk J. and Zadrożny S. (1995a): *Fuzzy queries in Microsoft Access.* — Proc. FUZZ-IEEE/IFES'95, Yokohama, Japan, Workshop *Fuzzy Database Systems and Information Retrieval*, Vol.2, pp.61–66.

Kacprzyk J. and Zadrożny S. (1995b): *FQUERY for Access: fuzzy querying for a Windows-based DBMS*, In: Fuzziness in Database Management Systems (P. Bosc and J. Kacprzyk, Eds.). — Heidelberg: Physica-Verlag, pp.415–433.

Kacprzyk J. and Zadrożny S. (1996): *A fuzzy querying interface for a WWW-server-based relational DBMS.* — Proc. IPMU'96, Granada, Spain, Vol.1, pp.19–24.

Kacprzyk J. and Zadrożny S. (1997): *Fuzzy queries in Microsoft Access*, In: Fuzzy Information Engineering Applications (D. Dubois, H. Prade and R.R. Yager, Eds.). — New York: Wiley, Vol.2, pp.223–232.

Kacprzyk J. and Zadrożny S. (1998): *Data mining via linguistic summaries of data: An interactive approach*, In: Methodologies for the Conception, Design and Application of Soft Computing (T. Yamakawa and G. Matsumoto, Eds.). — Proc. IIZUKA'98, Iizuka, Japan, FLSI, pp.668–671.

Kacprzyk J. and Zadrożny S. (1999): *On interactive linguistic summarization of databases via a fuzzy-logic-based querying add-on to Microsoft Access*, In: Computational Intelligence: Theory and Applications (B. Reusch, Ed.). — Heidelberg: Springer-Verlag, pp.462–472.

Kacprzyk J. and Zadrożny S. (2000a): *On combining intelligent querying and data mining using fuzzy logic concepts*, In: Recent Research Issues on the Management of Fuzziness in Databases (G. Bordogna and G. Pasi, Ed.). — Heidelberg, New York: Physica-Verlag (in press).

Kacprzyk J. and Zadrożny S. (2000b): *Data mining via fuzzy querying over the internet*, In: Knowledge Management in Fuzzy Databases (O. Pons, M.A. Vila and J. Kacprzyk, Eds.). — Heidelberg and New York: Physica-Verlag, pp.211–233.

Kacprzyk J. and Zadrożny S. (2000c): *Using fuzzy querying over the internet to browse through information resources*, In: Computing with Words (P.P. Wang, Ed.). — New York: Wiley (in press).

Kacprzyk J. and Ziółkowski A. (1986b): *Database queries with fuzzy linguistic quantifiers.* – IEEE Trans. Syst. Man Cybern., Vol.16, pp.474–479.

Liu Y. and Kerre E.E. (1998): *An overview of fuzzy quantifiers. (I) Interpretations.* — Fuzzy Sets Syst., Vol.95, pp.1–21.

Pons O., Vila M.A. and Kacprzyk J. (Eds.) (2000): *Knowledge Management in Fuzzy Databases.* — Heidelberg, New York: Physica-Verlag.

Rasmussen D. and Yager R.R. (1996): *Using summary SQL as a tool for finding fuzzy and gradual functional dependencies.* — Proc. IPMU'96, Granada, Spain, pp.275–280.

Rasmussen D. and Yager R.R. (1997a): *A fuzzy SQL summary language for data discovery*, In: Fuzzy Information Engineering: A Guided Tour of Applications (D. Dubois, H. Prade and R.R. Yager, Eds.). — New York: Wiley, pp.253–264.

Rasmussen D. and Yager R.R. (1997b): *Summary SQL – A fuzzy tool for data mining, Intelligent Data Analysis.* – An Int. J. 1 (Electronic Publication), URL-`http://www-east.elsevier.com/ida/browse/96-6/ida96-6.htm`.

Rasmussen D. and Yager R.R. (1999): *Finding fuzzy and gradual functional dependencies with summary SQL.* — Fuzzy Sets Syst., Vol.106, pp.131–142.

Saaty T.L. (1980): *The Analitic Hierarchy Process: Planning, Priority Setting, Resource Allocation.* — New York: McGraw-Hill.

Yager R.R. (1982): *A new approach to the summarization of data.* — Inform. Sciences, Vol.28, pp.69–86.

Yager R.R. (1988): *On ordered weighted avaraging operators in multicriteria decision making.* — IEEE Trans. Syst. Man Cybern., Vol.SMC–18, pp.183–190.

Yager R.R. (1989): *On linguistic summaries of data.* — Proc. IJCAI Workshop *Knowledge Discovery in Databases*, Detroit, pp.378–389.

Yager R.R. (1991): *On linguistic summaries of data*, In: Knowledge Discovery in Databases (G. Piatetsky-Shapiro and B. Frawley, Eds.). — Cambridge, MA: MIT Press, pp.347–363.

Yager R.R. (1994): *Linguistic summaries as a tool for database discovery.* — Proc. Workshop *Flexible Query-Answering Systems*, Roskilde University, Denmark, pp.17–22.

Yager R.R. (1995a): *Linguistic summaries as a tool for database discovery.* — Proc. Workshop *Fuzzy Database Systems and Information Retrieval at FUZZ-IEEE/IFES*, Yokohama, Japan, pp.79–82.

Yager R.R. (1995b): *Fuzzy summaries in database mining.* — Proc. 11th Conf. *Artificial Intelligence for Applications*, Los Angeles, USA, pp.265–269.

Yager R.R. (1996): *Database discovery using fuzzy sets.* — Int. J. Intell. Syst., Vol.11, pp.691–712.

Yager R.R. and Kacprzyk J. (1997): *The Ordered Weighted Averaging Operators: Theory and Applications.* - Boston: Kluwer.

Yager R.R. and Kacprzyk J. (1999): *Linguistic Data Summaries: A Perspective.* — Proc. IFSA'99 Congress, Taipei, Taiwan R.O.C., Vol.1, pp.44–48

Yager R.R. and Rubinson T.C. (1981): *Linguistic summaries of data bases.* — Proc. IEEE Conf. *Decision and Control*, San Diego, USA, pp.1094–1097.

Zadeh L.A. (1983): *A computational approach to fuzzy quantifiers in natural languages.* — Comp. Maths. Appl., Vol.9, pp.149–184.

Zadeh L.A. (1985): *Syllogistic reasoning in fuzzy logic and its application to usuality and reasoning with dispositions.* — IEEE Trans. Syst. Man Cybern., Vol.SMC-15, No.??, pp.754–763.

Zadeh L.A. and Kacprzyk J. (Eds.) (1992): *Fuzzy Logic for the Management of Uncertainty.* — New York: Wiley.

Zadeh L. and Kacprzyk J. (Eds.) (1999a): *Computing with Words in Information/Intelligent Systems 1. Foundations.* — Heidelberg, New York: Physica-Verlag.

Zadeh L. and Kacprzyk J. (Eds.) (1999b): *Computing with Words in Information/Intelligent Systems 2. Applications.* — Heidelberg, New York: Physica-Verlag.

Zadrożny S. and Kacprzyk J. (1995): *Fuzzy querying using the 'query-by-example' option in a Windows-based DBMS.* — Proc. EUFIT'95, Aachen, Germany, Vol.2, pp.733–736.

Zadrożny S. and Kacprzyk J. (1999): *On database summarization using a fuzzy querying interface.* — Proc. IFSA'99 World Congress, Taipei, Taiwan R.O.C., Vol.1, pp.39–43.

Zemankova and Kacprzyk J. (1993): *The role of fuzzy logic and management of uncertainty in building intelligent information systems.* — J. Intell. Inform. Syst., Vol.2, pp.311–317.