BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 58, No. 3, 2010
DOI: 10.2478/v10175-010-0034-2

Towards human consistent data driven decision support systems using verbalization of data mining results via linguistic data summaries

J. KACPRZYK* and S. ZADROŻNY

Systems Research Institute, Polish Academy of Sciences, 6 Newelska St., 01-447 Warsaw, Poland

Abstract. We present how the conceptually and numerically simple concept of a fuzzy linguistic database summary can be a very powerful tool for gaining much insight into the essence of data that may be relevant for a business activity. The use of linguistic summaries provides tools for the verbalization of data analysis (mining) results which, in addition to the more commonly used visualization e.g. via a GUI, graphical user interface, can contribute to an increased human consistency and ease of use. The results (knowledge) derived are in a simple, easily comprehensible linguistic form which can be effectively and efficiently employed for supporting decision makers via the data driven decision support system paradigm. Two new relevant aspects of the analysis are also outlined which was first initiated by the authors. First, following Kacprzyk and Zadrożny [1] comments are given on an extremely relevant aspect of scalability of linguistic summarization of data, using their new concept of a conceptual scalability that is crucial for large applications. Second, following Kacprzyk and Zadrożny [2] it is further considered how linguistic data summarization is closely related to some types of solutions used in natural language generation (NLG), which can make it possible to use more and more effective and efficient tools and techniques developed in this another rapidly developing area. An application of a computer retailer is outlined.

Key words: decision support system, data mining.

1. Introduction

Decision making is a "meta-problem" in all kinds of human activities, and a natural consequence has been that it has become an object of intensive research which has been done in many fields, including mathematics, economics, social sciences, cognitive sciences, etc., and also along different lines and perspectives. Notably, many powerful formal (mathematical) models have been proposed.

However, as the complexity of decision problems increases, the use of rigid formal models, that require much knowledge and information about the decision situation, goals, constraints, etc. may become difficult. An effective and efficient solution in such cases may be the use of a *decision support system* (DSS) – see, e.g., Alter [3], Holsapple and Whinston [4], Sprague and Watson [5], etc. which is not meant to replace the decision maker but support him or her. Historically, DSSs practically appeared in the mid-1960s with the development of IBM 360 and a wider use of distributed, time-sharing computing, and have been since that time a topic of intensive research and development.

One can distinguish the following basic types of DSSs (cf. Dan Power's: www.dssresources.com):

- Data driven,
- Communication driven and group DSSs,
- Document driven,
- Model driven,
- Knowledge driven,
- Web based and interorganizational.

Roughly speaking, except for the model driven DSSs, all other ones do not explicitly use models of decision situations in question but try to support the decision maker by giving him or her access to information and knowledge, facilitating communication with other agents involved, etc. This is considered to be a proper solution in many real decision making processes.

This work is mainly concerned with data driven DSSs that facilitate access to and manipulation of internal company data and sometimes external data, and may be based – from the low to the high level – first on simple file systems with query and retrieval tools, then data warehouses, and finally with On-line Analytical Processing (OLAP) or data mining tools. Some advanced, non-conventional data mining tools will be considered here.

In particular, we will consider how the use of Zadeh's computing with words (and perceptions) paradigm (cf. Zadeh and Kacprzyk [6]) through fuzzy linguistic database summaries, and indirectly fuzzy querying, can open new vistas in data driven DSSs due to an extensive use of natural language which is the only fully natural means of articulation and communication by the humans.

We discuss linguistic summarization of data sets in the sense of Yager [7], extended and presented in an imple-

*e-mail: kacprzyk @ibspan.waw.pl

mentable form by Kacprzyk and Yager [8], and Kacprzyk, Yager and Zadrożny [9]. In this approach a linguistic summary is derived as a linguistically quantified proposition, exemplified – when the data in question concern employees – by "most of the employees are young and well paid", with which a degree of validity is associated. Basically, in this approach, if we have a relational database, for instance containing data on employees (age, sex, salary, etc.), then – if we choose attribute "age" as an object of interest - a linguistic summary of a data set consists of: a summarizer S (e.g. young), a quantity in agreement Q (e.g. most), and the truth value (validity) T – e.g. 0.7, and may be exemplified by "T(most of employees are young)=0.7". Moreover, if we have an additional qualifier (e.g., "well paid"), then a linguistic summary can be "T(most of well paid employees are young)=0.8". Basically, for a set of data D, we can hypothesize any appropriate summarizer S and any quantity in agreement Q, and the assumed measure of truth will indicate the truth of the statement that Q data items satisfy the statement (summarizer) S.

Notice that we use here highly imprecise descriptions both for the value of an attribute chosen and the number of such employees but the use of precise terms makes obviously no sense since we wish to just provide an extremely simple linguistic summary that could grasp the very meaning of a possibly huge data set. Notice also that we deal here with a fuzzy approach to linguistic data summarization though many other approaches, notably those based on statistics, are known in the computational linguistics and natural language processing communities, and can be found in an abundant literature. However, in our approach the problem is to deal with the imprecision of meaning and we think that fuzzy logic can provide simple and efficient tools and techniques for this purpose.

A fully automatic derivation of the linguistic summaries is computationally infeasible. Thus, we advocate an interactive approach instead, which requires some high-level, abstract representation of the summaries, which may be instantiated in various ways so as to properly represent user's idea of an interesting summary. This makes the concept of a *protoform* in the sense of Zadeh highly relevant. A protoform is defined as an abstract prototype, that is, with respect to those summaries given above:

"Most R's are S"

or

"Most *BR*'s are S"

where R means "records", B is a condition, and S is a query.

Protoforms can obviously form a hierarchy, so that we can define higher level (more abstract) protoforms, for instance replacing "most" by a general linguistic quantifier Q, obtaining, respectively: "*QR*'s are S" and "*QBR*'s are S".

Clearly, the protoforms are a powerful conceptual tool because we can formulate many different types of linguistic summaries in a uniform way, and devise a uniform and universal way to handle different linguistic summaries. Therefore, the use of protoforms is very relevant, and also contributes to an increased *conceptual scalability* of linguistic data summarization introduced by Kacprzyk and Zadrożny [10,11] as the simplicity and intuitive appeal of the protoforms used in the context of linguistic data summaries make them applicable to data sets of any size. Even if the size of a data set increases, the very essence of a particular protoform just catches the contents of the data set in a user comprehensible form.

Another aspect, which is relevant in our context, is whether one can also use in the process of linguistic summarization of data sets some other tools and techniques known in other areas, for which new, more effective and efficient approaches and methods are being proposed. If so, one could expect that we can use those new results for our ultimate benefit, that is, to make linguistic data summarization applicable to large problems. In this perspective, it was shown in recent papers [10, 11] that the linguistic data summarization as meant in this paper and viewed from the perspective of linguistic summaries as protoforms, is related to natural language generation (NLG), notably in the "numbers to words" path (cf. Reiter and Dale [12]). Basically, it can be shown that linguistic summarization is strongly related to template based NLG systems. Moreover, our approach to linguistic summarization can be viewed, from some perspective, as a simple phrase based system. In general, relations between our protoform based approach to linguistic data summarization, and modern approaches and solutions employed in the field of natural language generation (NLG), as indicated by Kacprzyk and Zadrożny [2, 11], are very strong and promising.

The linguistic data summaries can provide some highly human consistent tools for extracting knowledge from relevant, usually large, data sets. The knowledge thus obtained is extremely well comprehensible by the human user because it is in a simple natural language form. This can be decisive for an easy implementation of a data driven decision support system. In fact, this marvelous property of linguistic data summaries have been one of main reasons for the success of an implementation for a sales decision support for a small computer retailer [13–15]. Basically, the system provides simple linguistic summaries on relations between some selected (by the user) attributes. They can be exemplified, for a relation between the "group of products" and "commission" by: "very few sales of printers are with a high commission", and may be useful for decision making.

The real implementation of our approach mentioned above stands in our case for a proof of an experimental evaluation which is presumably the main challenge and problem in virtually all kinds of computational linguistics and natural language processing related works as can be seen from contributions at main scientific events in those areas.

To summarize, through an extensive use of natural language via the verbalization of data mining results we have been able to attain an extremely high human consistency that may be crucial for a successful use of a data driven decision support system as has been shown by our implementation. Moreover, results from a new, rapidly developing area of natural language generation can provide new inspiration, concepts and tools.

Notice that though visualization has attracted so far most attention of researchers, maybe by following a well known statement that "one picture is worth thousand words", one should bear in mind that visualization implies a necessity to look at a visual display which can be in many applications a limiting factor because attention may be too much distracted, like in – for example – many military applications, intelligent transportation systems, etc. Verbalization, i.e. the presentation of results in (usually spoken, synthetized) natural language statements, distracts usually attention to a much lesser degree, and may often be desirable.

We will discuss in the consecutive parts issues related to decision support systems, notably non-model driven ones, and emphasize the ones based on data mining. Then, we will present linguistic data summaries with the use of a fuzzy querying interface for their derivation, and finally discuss relations to natural language. We will illustrate our discussion on an implementation for a small computer retailer.

2. Decision making and decision support systems

Decision making, due to its primordial importance and universal relevance, has been a subject of analysis, and the research since the ancient times. In recent times it has concentrated on attempts for a formal, mathematical analysis and the development of mathematical models to describe the decision making setting, decision makers' intentions, solutions, etc. Both descriptive and prescriptive, involving single and multiple criteria and decision makers, dynamics, etc. approaches have been proposed and used.

There is a departure from this idealistic paradigm in modern approaches to real world decision making (cf. Wierzbicki, Makowski and Wessels [16]) in which *good decisions* (not *optimal* as in most traditional approaches) are sought. A *decision making process* is considered which involves basically:

- Use of own and external knowledge,
- Involvement of various "actors", aspects, etc.
- Individual habitual domains,
- Non-trivial rationality,
- Different paradigms, when appropriate.

A good example of such a decision making process is Peter Checkland's [17, 18] so-called *deliberative decision making* (which is an important element of his *soft approach to systems analysis*). The essence of deliberative (soft) decision making may be subsumed as follows: to solve a complex real world decision making problem we should: perceive the whole picture, observe it from all angles (actors, criteria, ...), and find a *good* decision using *knowledge* and *intuition*.

Modern approaches to the decision making process assume as its crucial elements:

- Recognition,
- Deliberation and analysis,
- Gestation and enlightenment (the so-called "eureka!", "aha" effects),
- Rationalization,
- Implementation.

Moreover, it is commonly emphasized that the process of arriving at a good decision:

- Is to be based not only on data and information, but on knowledge and human specific characteristics (intuition, attitude, natural language for communication and articulation, ...),
- Needs number crunching, at which the computers are good, but also more "delicate" and sophisticated "intelligent" analyses, at which the human being is better,
- Should rely in realistic settings on computer systems, and on a synergistic human-computer interaction, using tools and techniques better suited to human cognitive capabilities, notably using graphical displays, i.e. *visualization*, and (quasi)natural language, i.e. *verbalization* during the problem formulation, solution, displaying of results, etc.

Therefore, to effectively and efficiently solve real world decision making problems, we should be supported by *decision support systems* (DSSs) – see, e.g. [4, 9] and [5]. They would make possible to tackle: ill/semi/un-structured questions and problems, a need for non-routine, one of a kind answers; to provide a flexible combination of analytical models and data; to handle various kinds of data (e.g. numeric, textual, multimedia); to provide "what if …" analyses; support various decision making styles; to provide tools to use both *explicit* (expressed in words or numbers, and shared as data, equations, specifications, documents, and reports, that can be transmitted between individuals and formally recorded) and *tacit* (personal, hard to formalize, and difficult to communicate or share with others) knowledge.

As mentioned in Introduction, one can distinguish the following basic types of DSSs (cf. Dan Power's classification: www.dssresources.com):

- Data driven,
- Communication driven and group DSSs,
- Document driven,
- Model driven,
- Knowledge driven,
- Web based and interorganizational.

Roughly speaking, data driven DSSs emphasize access to and manipulation of internal company data and sometimes external data, and may include simple file systems with query and retrieval tools, data warehouses, and finally On-line Analytical Processing (OLAP) or data mining tools. Communication driven DSSs use network and communications technologies to facilitate collaboration and communication. Group GDSSs are interactive systems that facilitate solution of unstructured problems by a group of decision-makers. Document driven DSSs include storage and processing technologies for a full document (numeric, textual and multimedia) retrieval and analysis. Model driven DSSs emphasize access to and manipulation of a model, e.g., statistical, optimization, simulation. Knowledge driven DSSs are interactive systems with specialized problem-solving knowledge about a particular domain. Web based DSSs deliver decision support related information and/or tools using a "thin-client" Web browser, the TCP/IP protocol, etc.

In this paper we concentrate on the data driven DSSs, and in particular show how the use of Zadeh's computing with words and perception paradigm (cf. Zadeh and Kacprzyk [6]) through fuzzy linguistic database summaries, implemented via fuzzy querying, can open new vistas in data driven DSSs (and also, to some extent, in knowledge driven and Web based DSSs) by its simplicity and a high scalability mainly due to an extensive use of natural language which is the only fully natural means of articulation and communication by the humans.

The basic role of a data driven DSS is to help decision makers make rational use of (vast) amounts of data that exist in their environment (e.g. a company or institution) to find some useful, relevant, nontrivial dependencies. One of interesting and promising approaches meant for these purposes is to derive linguistic summaries of a set of data (database). Here we discuss linguistic summarization of data sets in the sense of Yager [7, 19] (for some extensions and other related issues, see, e.g., [8, 9, 20–24]. In our context linguistic data summaries are derived as linguistically quantified propositions, exemplified – when the data in question concern employees – by "most of the employees are young and well paid", with which a degree of validity is associated.

This paper is based on [25–29] and Zadrożny and Kacprzyk's [14] idea of an interactive approach to linguistic summaries, i.e. assuming that an interaction with the user is practically needed for the determination of a class of summaries of interest. This is implemented via Kacprzyk and Zadrożny's [30, 31] fuzzy querying add-on to Microsoft Access.

Then, following and extending Kacprzyk and Zadrożny [10], we show that by relating various types of linguistic summaries to various fuzzy queries, we can arrive at a hierarchy of Zadeh's *protoforms* of linguistic data summaries which are conceptually very powerful by providing an unified structural form of even the most complicated linguistic summaries.

By using natural language to present (verbalize) the essence of data and relations of interest we attain a high, maybe an ultimate human consistency because natural language is the only fully natural means of articulation and communication of a human being. Moreover, through natural language we attain an ultimate scalability as natural language can express in a comprehensive way information no matter how large the data set is; cf. [1] in which the concept of a *conceptual scalability* has been introduced as a complement to the technical scalability normally considered.

Another important aspect is whether linguistic data summaries are related to some other well established techniques, and in this respect Kacprzyk and Zadrożny [2, 11] have indicated that they directly correspond to some specific, so-called template based, techniques of natural language generation (cf. [12]), but extend those traditional techniques by making it possible to account for the inherent imprecision of natural language. We will consider this issue in more detail using recent results obtained by Kacprzyk and Zadrożny [2].

One should also notice that we deal in this paper with the use of fuzzy logic to linguistic data summarization and do not consider other powerful approaches, notably those based on statistics. We do not provide a comparison with those approaches, which might have been a very interesting and challenging task, and show that our approach is proper in the problem considered, and hopefully in a whole array of other problems, by presenting its implementability in a real decision support system of a computer retailer. The perspective and attitude adopted in our work is justified to some extent by results of [32] who analyze the power of various approaches to prediction in real world, business setting and clearly advocate some more human consistent approaches that use "softer" tools and techniques.

3. Linguistic data summaries

We start with a basic approach to the linguistic summarization of data sets proposed by Yager [7], and then presented in a more advanced, and implementable form by Kacprzyk and Yager [8], and Kacprzyk, Yager and Zadrożny [9]. We have:

- V is a quality (attribute) of interest, e.g. salary in a database of workers,
- $Y = \{y_1, \ldots, y_n\}$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 $y_i \in Y$,
- $D = \{V(y_1), \ldots, V(y_n)\}$ is a set of data (the "database" in question).

A *linguistic summary* of a data set D consists of:

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

as, e.g., "T(*most* of employees are *young*)=0.7". The truth T may be meant in a more general sense, e.g. as validity or, even more generally, as some quality or goodness of a linguistic summary.

Notice that we consider here some specific, basic form of a linguistic summary that concerns sets of numeric values only. For some conceptually different approaches, which have been proposed in the fuzzy logic related areas, cf. [20–24, 33]. For some other, non-fuzzy-logic based approaches, see for instance [33].

The summarizer S is a linguistic expression semantically represented by a fuzzy set. For instance "young" may be represented as a fuzzy set in the universe of discourse as, e.g., $\{1, 2, \ldots, 90\}$, i.e. containing possible values of the human age, and "young" could be given as, e.g., a fuzzy set with a non-increasing 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. This kind of a summarizer exemplified by "young" can clearly be extended to, e.g, "*young* and *well paid*". More sophisticated, and more interesting summarizers (concepts) as, e.g.: of productive workers, difficult orders, etc. defined by a 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. will be discussed later.

The (linguistic) quantity in agreement, Q, is an indication of the range of data satisfying the summarizer, and is assumed to be a linguistic term represented by a fuzzy set, of a relative type as, e.g., "a few", "more or less a half", "most", "almost all", etc., equated with the so-called fuzzy linguistic quantifiers (cf. Zadeh [34]) that can be handled by fuzzy logic.

The calculation of the truth (or, more generally, validity) of the linguistic summary considered is equivalent to the calculation of the truth value of a corresponding linguistically quantified statement (e.g., "*most* of the employees are *young*"). This can be calculated by using two most relevant techniques: Zadeh's [34] calculus of linguistically quantified statements (cf. Zadeh and Kacprzyk [6]) or Yager's [7] OWA operators (cf. Yager and Kacprzyk [35]). In what follows we briefly remind the basics of these two techniques.

A linguistically quantified proposition, e.g., "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)". Property F and importance B are fuzzy sets in Y, and a (proportional, non-decreasing) linguistic quantifier Q is assumed to be a fuzzy set in [0,1] as, e.g.

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

Then, due to Zadeh [34]

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

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

An OWA operator (Yager [7], Yager and Kacprzyk [35]) of dimension p is a mapping $O : [0,1]^p \to [0,1]$ if associated with O is a weighting vector, $W = [w_1, \ldots, w_p]^T$, $w_i \in [0,1], w_1 + \cdots + w_p = 1$, and

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

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

The OWA weights may be found from the membership function of Q due to (cf. Yager [36]):

$$w_i = \mu_Q \left(\frac{i}{p} \right) - \mu_Q \left(\frac{i-1}{p} \right) \qquad \text{for } i = 1, \ldots, p. \tag{5}$$

The OWA operators can model a wide array of aggregation operators (including linguistic quantifiers), from $w_1 = \ldots = w_{p-1} = 0$ and $w_p = 1$ which corresponds to "all", to $w_1 = 1$ and $w_2 = \ldots = w_p = 0$ which corresponds to "at least one", through all intermediate situations, and that is why they are widely employed.

One can also extend the OWA operator to include importance qualification of the particular pieces of data. Suppose that with the data $A = [a_1, \ldots, a_p]$, a vector of importances $V = [v_1, \ldots, v_p]$, such that $v_i \in [0,1]$ is the importance of $a_i, i = 1, \ldots, p$, $v_1 + \cdots v_p = 1$, is associated. Then, for an *ordered weighted averaging operator with importance qualification*, denoted O_I , Yager [36] proposed that, first, some redefinition of the OWA's weights w_i's into $\overline{w}_i'$s is performed, and (4) becomes

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

where

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

where u_k is the importance of b_k, i.e. of the k-largest element of A. For some more advanced issues related to the choice and tuning of weights of the OWA operators, cf. Zadrożny and Kacprzyk [37].

The basic validity criterion of the truth of a linguistically quantified statement given by (2) and (3) is certainly the most natural and important but it does not grasp all aspects of a linguistic summary. Some other, additional quality criteria have been proposed in the literature, starting from some measure of informativeness in the source Yager's [7] paper, through some measures given by George and Srikanth [38], to a comprehensive set of measures given by Kacprzyk and Yager [8], and Kacprzyk, Yager and Zadrożny [9] who have proposed:

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

Due to lack of space, we will not discuss these measures referring the interested readers to the papers cited.

Now, denoting the above degrees of validity as T_1, T_2, T_3, T_4, T_5, with the respective weighs, w_1, w_2, w_3, w_4, w_5, assigned (with values from the unit interval, the higher, the more important such that $\sum_{i=1,2,\ldots,5} w_i = 1$), the (total) degree of validity, T, of a particular linguistic summary is defined as the weighted average of the above 5 degrees of validity, i.e.:

$$T = T(T_1, T_2, T_3, T_4, T_5; w_1, w_2, w_3, w_4, w_5) = \sum_{i=1,2,\ldots,5} w_i T_i \tag{8}$$

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

$$S^* = \text{arg} \max_S \sum_{i=1,2,\ldots,5} w_i T. \tag{9}$$

The weights, $w_1, \ldots, w_5$, may be predefined or elicited from the user, e.g, using Saaty's [39] AHP (analytical hierarchy process) technique.

The linguistic summarization meant as the solution of (9) may be numerically difficult in general, hence is in principle not well scalable in the traditional sense but, using the concept of cognitive (perceptual) scalability introduced by Kacprzyk and Zadrożny [1], it may be said to be totally conceptually (perceptually) scalable because it is comprehensible to a human being no matter what size of the data set is. This is a direct result of, on the one hand, the use of natural language, and – on the other hand – of a simple and intuitively appealing form of a linguistic summary which basically says what most of the data exhibit, i.e. what *usually happens* (holds).

A fully automatic determination of a best linguistic summary, i.e. the solution of (9) may be therefore infeasible in practice, and therefore Kacprzyk and Zadrożny [25, 26, 28] proposed an *interactive approach* with *user assistance* in the definition of summarizers, by the indication of attributes and their combinations of interest. This proceeds via a user interface of a *fuzzy querying* add-on. Basically, the queries (referring to summarizers) allowed are:

- *simple* as, e.g., "salary is *high*",
- *compound* as, e.g., "salary is *low* AND age is *old*",
- *compound (with quantifier)*, as, e.g., "*most* of {salary is *high*, age is *young*, ..., training is *well above average*}".

In Kacprzyk and Zadrożny [26–31], a conventional DBMS is used, and a fuzzy querying tool FQUERY for Access is developed to allow for queries with fuzzy (linguistic) elements of the "simple", "compound" and "compound with quantifier" types. This fuzzy querying system (add-in) has been developed for Microsoft Access® but it is clearly applicable to any DBMS. The main problems to be solved are here: (1) how to extend the syntax and semantics of the query, and (2) how to provide an easy way of eliciting and manipulating those terms by the user.

FQUERY for Access is embedded in the native Microsoft Access's environment as an add-in. All the code and data is put into a database file, a *library*, installed by the user. Definitions of attributes, linguistic terms etc. are maintained in a dictionary (a set of regular tables), and a mechanism for putting them into the Query-By-Example (QBE) sheet (grid) of the Microsoft Access' interface is provided. Linguistic terms are represented within a query as parameters, and a query transformation is performed to provide for their proper interpretation during the query execution.

FQUERY for Access makes it possible to use various linguistic (fuzzy) terms in queries: fuzzy values, exemplified by *low* in "profitability is *low*", fuzzy comparators, 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", where 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.

Fuzzy values are defined as fuzzy sets on the interval $[-10, +10]$, which is treated as a universal universe of discourse, making fuzzy values applicable to any numerical attribute of the queried data. Then, *the matching degree md* $(\cdot,\cdot)$ of a simple condition referring to attribute *AT* and fuzzy value *FV* against a record (tuple) t is calculated by:

$$md(AT = FV, t) = \mu_{\text{FV}}(\tau(t.AT)), \tag{10}$$

where *t.AT* is the value of attribute *AT* at the tuple t, μ_{FV} is the membership function defining fuzzy value *FV*, and τ: $[\text{LL}_{AT},\text{UL}_{AT}]\rightarrow[-10,10]$ is the mapping from the domain of *AT* onto $[-10,10]$, securing the applicability of a fuzzy value to any attribute, mentioned above. Thus, τ which makes it possible to treat all attributes' domains as ranging over the interval $[-10,10]$. For simplicity, it is normally assumed, also here, that the membership functions of fuzzy values are trapezoidal as in Fig. 1 and τ is assumed linear.

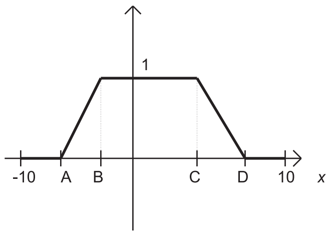

Fig. 1. An example of the membership function of a fuzzy value

Linguistic quantifiers provide for a flexible aggregation of simple conditions. In FQUERY for Access the fuzzy linguistic quantifiers are defined in Zadeh's [34] sense; cf. *most* given as (1). They may be interpreted and manipulated either using original Zadeh's approach or via the OWA operators. The matching degree, *md*$(\cdot,\cdot)$, for the query "Q of N conditions are satisfied" for record t is computed using (2), where F is interpreted as a fuzzy set of conditions, with the membership degree equal the matching degree of given condition for tuple t. If particular conditions are assigned different importance degrees, then the formula (3) is used .

In FQUERY for Access queries containing fuzzy terms are still syntactically correct Access's queries through the use of parameters. Access represents the queries using SQL, and the parameters make it possible to embed references to fuzzy terms in a query. For example, a parameter like: [FfA_FV *fuzzy value name*] will be interpreted as a fuzzy value, while[FfA_FQ *fuzzy quantifier name*] will be interpreted as a fuzzy quantifier. Fuzzy terms are defined using the toolbar provided by FQUERY for Access, stored internally in the dictionary maintained in the system.

When the user initiates the execution of a query it is automatically transformed by appropriate FQUERY for Access's routines 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 usually.

Clearly, fuzzy queries directly correspond to summarizers in linguistic summaries which was first formally shown by Kacprzyk and Zadrożny [25]. Thus, the derivation of a linguistic summary may proceed in an interactive (user assisted) way as follows:

- the user formulates a set of linguistic summaries of interest (relevance) using the fuzzy querying add in,
- the system retrieves records from the database and calculates the validity of each summary adopted, and
- some best (most appropriate) linguistic summary is chosen.

Therefore, we can restate the linguistic summarization in the fuzzy querying context. First, (2) may be interpreted as:

$$\text{“Most records match query } S\text{”}, \qquad (11)$$

where S replaces F in (2) since we refer here directly to the concept of a summarizer (of course, S is in fact the whole condition, e.g., price = *high*, while F is just the fuzzy value, i.e. *high* in this condition; this should not lead to confusion).

Similarly, (3) may be interpreted as:

$$\text{“Most records meeting conditions } B \text{ match query } S\text{”} \qquad (12)$$

and notice that B, which may be treated in fact as another query, corresponds to a *filter* and (12) claims that *most* records passing through B match query S; clearly, this may be to a degree from [0,1] in our context. Still another interpretation of (12) may be expressed as follows: “Most of the records *among* those satisfying a query B do satisfy also the query S”. Due to the fact that both B and S are, in general, fuzzy the meaning of (12) cannot be replaced with a classical logical combination of them.

Looking at (11) and (12), i.e. the user's interest and intention as to linguistic data summaries put in the context of database querying, Kacprzyk and Zadrożny [10] showed that the concept of Zadeh's [40] *protoform* is highly relevant. A protoform (prototypical form) is defined as an more or less abstract prototype of a linguistically quantified proposition, that takes in the most abstract case the following form (where Q denotes any linguistic quantifier):

$$\text{“}QR\text{'s are } S\text{”} \qquad (13)$$

or

$$\text{“}QBR\text{'s are } S\text{”}. \qquad (14)$$

A protoform represents a class of linguistic summaries, which may be obtained via instantiation of abstract symbols possibly appearing in this protoform and denoting any linguistic quantifier, summarizer or qualifier. As such it may be used to represent the preferences of the user as to the form of the linguistic summaries sought. The protoforms may be more or less general, and more abstract protoforms correspond to cases in which we assume less about summaries sought, with two limit cases: (A) we assume a totally abstract protoform (13) or (14), and (B) we assume all elements of a protoform to be given (instantiated). In case A data summarization will be extremely time consuming, as the search space may be enormous (all abstract symbols may be instantiated in many different ways), but may produce interesting, unexpected views on data. In case B the user is in fact guessing a good candidate summary but the evaluation is simple, equivalent to the answering of a (fuzzy) query; case B refers to the concept of *ad hoc queries* (Anwar et al., 1992).

In Table 1 a useful classification of linguistic summaries into 5 basic types corresponding to increasingly more abstract protoforms is shown.

Briefly speaking, Type 1 summaries may be easily derived by a simple extension of fuzzy querying via FQUERY for Access. The user has to construct a query, a candidate summary, and the derivation module has just to find the fraction of rows matching this query and a linguistic quantifier best denoting this fraction. A Type 2 summary is a straightforward extension of Type 1 by adding a fuzzy filter. Type 3 summaries are concerned with the determination of typical (exceptional) values of an attribute. So, query S consists of only one simple condition with the attribute whose typical (exceptional) value is sought, the “=” relational operator and a placeholder for the value sought. A Type 4 summary may produce typical (exceptional) values for some, possibly fuzzy, subset of rows. Type 5 summaries represent the most general form considered here: fuzzy rules describing dependencies between specific values of particular attributes. The summaries of Type 1 and 3 have been implemented (cf. Kacprzyk and Zadrożny [10, 26, 27]) as an extension to FQUERY for Access. Type 5 summaries can be generated by analogy to *association rules* and employing algorithms for mining them, or by using genetic algorithms to search the space of possible summaries (cf. Kacprzyk and Zadrożny [10]).

Table 1
Classification of linguistic summaries

Type	Given	Sought	Remarks
1	S	Q	Simple summaries through ad-hoc queries
2	SB	Q	Conditional summaries through ad-hoc queries
3	$QS^{structure}$	S^{value}	Simple value oriented summaries
4	$QS^{structure}B$	S^{value}	Conditional value oriented summaries
5	Nothing	SBQ	General fuzzy rules

where $S^{structure}$ denotes that attributes and their connection in a summary are known but the (fuzzy) values of these attributes are missing, while S^{value} denotes these missing (fuzzy) values.

4. Verbalization via linguistic data summaries and natural language generation (NLG)

As indicated in previous sections, linguistic data summaries may be very effective and efficient for the verbalization of results of all kinds of data analyses and data mining. One may immediately notice their strong resemblance to natural language generation (NLG) but this path was not explored so far too much. Maybe the first indication was given by Kacprzyk, Zadrożny and Wilbik [41] in which a reference to an NLG based approach to the linguistic summarization of time series, the SumTime project at the University of Aberdeen, UK (cf. Portet et al. [42] or Sripada, Reiter and Davy [43]) was made. Later, Kacprzyk and Zadrożny [11] more explicitly suggested a relation to NLG, and further elaborated on that in Kacprzyk and Zadrożny [2]).

Basically, natural language generation (NLG) – which is part of natural language processing (NLP) or, more generally, computational linguistics – is concerned with how one can automatically produce high quality natural language text from computer-internal representations of information which is not in natural language. In the case of linguistic summaries as considered here, this is the "numbers to words" path.

NLG may be viewed from many perspectives (cf. Reiter and Dale 12]) For our purposes it may be expedient to consider independently the tasks of generation and the process of generation. One can identify three types of tasks:

- text planning,
- sentence planning, and
- surface realization.

Text planners select what information to include in the output, and use it to form a proper text structure; sentence planners organize the content of sentences, notably order its parts, and surface realizers convert sentence sized chunks of representation into grammatically correct sentences.

In the context of linguistic data summarization, we have mainly considered text planning due to an explicit use of protoforms of linguistic summaries as fixed and specified (structurally), and by assuming the purpose of a protoform based linguistic summarization to determine appropriate linguistic values of the linguistic quantifier, qualifier and summarizer. However, if a protoform were considered in a "meta-sense", i.e. when the summarizer concerns the linguistically defined values of a compound concept, like "productivity" in a personnel database, with productivity described by an ordered (e.g. through importance assignment) list of criteria, or a hierarchical representation of a protoform were assumed (both would be more realistic!), then we would end up with some sort of sentence planning. Finally, the use of protoform based linguistic summaries precludes the use of surface planning in the strict sense. A solution might be eventually to develop different kinds of protoforms, notably not explicitly related to usuality, one of the most important modalities but to other modalities as well, but this is not trivial. So, the use of the sentence planning and surface realization would presumably produce qualitatively new linguistic summaries, with a larger expressive power, but it is not clear how to accommodate these tasks in Yager's concept of a linguistic summary, and our heavily protoform based approach.

Other classifications of generation tasks may be used but they do not essentially change the relations between linguistic summarization and NLG. Due to the use of protoforms, almost all tasks are simple due to a predefined structure of summaries but, to make a full use of the power of NLG tools, one should presumably devise other, richer types of protoforms. Another viable alternative is to use an interactive human – computer interface, as actually employed in our implementation of linguistic summaries.

Generation processes can be classified due to their sophistication and expressive power, starting with inflexible canned methods and ending with maximally flexible feature combination methods. The widely used canned text systems, just printing strings of words without any change (error messages, warnings, letters, etc.), are not interesting for us.

The template based approach is used mainly for multiple sentence generation, particularly when texts are regular in structure such as stock market reports. In principle, our approach to linguistic data summaries is similar in spirit to template based systems. One can say that Zadeh's protoforms can be viewed as playing a similar role to templates. However, there is an enormous difference as the protoforms are much more general and may represent such a wide array of various "templates" that maybe it would be more proper to call them "meta-templates". A interesting extension of our linguistic summarization might be to follow the multisentence path, cf. McKeown's [44] idea of dynamically nesting instances of some paragraphs, but so far it is not clear how one can extend the simple one sentence, protoform based structure of summaries adopted in our approach to this case.

Phrase based systems employ generalized templates, and a phrasal pattern is first selected to match the top level of the input and then each part of the pattern is expanded into a more specific phrasal pattern that matches subparts of the input, with the phrasal pattern replaced by one or more words. Such systems may be powerful but are very hard to build because of difficulties in a correct specification of phrasal interrelationships. It seems that our approach to linguistic summarization can be viewed, from some perspective, as a simple phrase based system. It should be also noted that since protoforms may form hierarchies, we can imagine that both the phrase and its subphrases can be properly chosen protoforms. The calculi of linguistically quantified statements can be extended to handle such a hierarchic structure of phrases (statements) though, at the semantic level, the same difficulties remain as in the NLG approach, i.e. an inherent difficulty to grasp the essence of multisentence summaries with their interrelations. We think that Zadeh's protoforms, but meant in a more general sense, for instance as hierarchical protoforms or "meta-protoforms", or even conceptually different protoforms, can make the implementation of a phrase based NLG system in our context viable.

Feature-based systems represent some extreme of the generalization of phrases. Each possible minimal alternative of expression is represented by a single feature. Generation proceeds by the incremental collection of features appropriate for each part of the input until the sentence is fully determined. Though their idea is very simple as any distinction in language is defined as a feature, analyzed, and added to the system, but unfortunately there is a tremendous difficulty in maintaining feature interrelationships and in the control of feature selection. It is not at all clear how our linguistic data summaries can be used within those systems.

There are other relevant aspects too. An extremely relevant issue, maybe a prerequisite for implementability, is domain modeling. The main difficulty is that it is very difficult to link a linguistic generation system to a knowledge base or data base originally developed for some nonlinguistic purpose due to a possible considerable mismatch. The construction of appropriate taxonomies or ontologies can be of much help. So far, in our approach, domain knowledge is dealt with via the specification of appropriate protoforms which are comprehensible or traditionally used (e.g. as structures of business reports) in a specific domain. Domain knowledge plays also an important role at the lower level of semantic interpretation (definition) of the linguistic terms. For some solutions we can refer to our approach (Kacprzyk and Zadrożny [45]) in which the use of ontologies providing the conceptualization of both the process itself and the domain of the decision making problem was shown for consensus reaching.

To summarize this short exposition of relations between our protoform based approach to the linguistic data summarization, and modern tools and techniques available in NLG, on the one hand, we can find much inspiration from recent developments in NLG, notably by showing intrinsic relations of Zadeh's protoforms to (meta-)templates or even simple phrase based systems, and in the adjusting of protoforms to domain specificity by employing some sentence and text planning tools.

From an implementation point of view since in NLG there is much commercial and open source software available, the user can find a proper software package that can be decisive for real world applications.

From the point of view of this paper, there is another crucial aspect, already mentioned in Introduction. Namely, linguistic data summaries provide knowledge in the form which is easily comprehensible by the human user because it is expressed in natural language, and is fully conceptually scalable. This is well illustrated with the practical success of our implementation for a sales decision support for a small computer retailer (cf. Kacprzyk and Strykowski [13]). We will now briefly show the very essence of how linguistic data summaries are used in the former implementation.

5. An example: linguistic data summaries to support sales decision making of a computer retailer

The proposed data summarization procedure was implemented in a data driven decision support system to support sales decision making of a computer retailer in Southern Poland (cf. Kacprzyk and Strykowski [13], Kacprzyk and Zadrożny [10, 28, 46]). The verbalization of data mining results provided by linguistic summaries have proven to be very useful. We will now briefly present the main idea of this implementation.

Though the database is large, its basic structure, which is relevant for our presentation, may be limited to its table shown in Table 2.

The derivation of the summaries is preceded by a dialogue with the user in which some parameters concerning mainly: definition of attributes and the definition of how the results should be presented. The results obtained in a real-life application are shown in the tables to follow.

Clearly, we have shown some most valid and more interesting linguistic summaries which may give the user much insight into relations between the attributes chosen, and are simple and human consistent.

Notice that these summaries concern data from the company's own database. However, companies operate in an environment (economic, climatic, social, etc.) which can be crucial. A notable example may here be the case of climatic data that can be fetched from some sources, paid or even free, mostly via the Internet. The inclusion of such data may be implemented as shown in Kacprzyk and Zadrożny [27, 29].

Table 2
Basic 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) x 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
About $1/2$ of sales of network elements is with a high commission
About $1/2$ of sales of computers is with a medium commission
Much sales of accessories is with a high commission
Much sales of components is with a low commission
About $1/2$ of sales of software is with a low commission
About $1/2$ of sales of computers is with a low commission
A few sales of components is without commission
A few sales of computers is with a high commission
Very few sales of printers is with a high commission

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

Summary
About 1/3 of sales of computers is by the end of year
About 1/2 of sales in autumn is of accessories
About 1/3 of sales of network elements is in the beginning of year
Very few sales of network elements is by the end of year
Very few sales of software is in the beginning of year
About 1/2 of sales in the beginning of year is of accessories
About 1/3 of sales in the summer is of accessories
About 1/3 of sales of peripherals is in the spring period
About 1/3 of sales of software is by the end of year
About 1/3 of sales of network elements is in the spring period
About 1/3 of sales in the summer period is of components
Very few sales of network elements is in the autumn period
A few sales of software is in the summer period

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
Much sales on Saturday is about noon with a low commission
Much sales on Saturday is about noon for bigger customers
Much sales on Saturday is about noon
Much sales on Saturday is about noon for regular customers
A few sales for regular customers is with a low commission
A few sales for small customers is with a low commission
A few sales for one-time customers is with a low commission
Much sales for small customers is for non-regular customers

Table 6
Linguistic summaries expressing relations between the attributes: group of products, time of sale, temperature, precipitacion, and type of customers

Summary
Very few sales of software in hot days to individual customers
About 1/2 of sales of accessories in rainy days on weekends by the end of the year
About 1/3 of sales of computers in rainy days to individual customers

For instance, if we are interested in relations between group of products, time of sale, temperature, precipitation, and type of customers, the best linguistic summaries (of both our “internal” data from the sales database, and “external” meteorological data from an Internet service) are as shown in Table 6.

Notice that the use of external data gives a new quality to possible linguistic summaries. It can be viewed as providing a greater adaptivity to varying conditions because the use of free or inexpensive data sources from the Internet makes it possible to easily and quickly adapt the form and contents of summaries to varying needs and interests. And this all is practically at no additional price and effort. A more elaborate concept of a decision support system taking into account an information context of the decision making process has been proposed recently by Kacprzyk and Zadrożny [47].

This concludes our very short exposition of the use of linguistic summaries in a data driven decision support system. As it can be seen, this solution – that is an example of verbalization of results – is conceptually simple and provides a new quality by providing an extremely human consistent insight to the essence of data and relations which could help to a considerable extent to make decisions.

6. Concluding remarks

In this paper we presented how the conceptually and numerically simple concept of a fuzzy linguistic database summary, viewed as a means for the verbalization of results of data analyses, data mining, knowledge discovery etc., can be a very powerful and human consistent tool for gaining insight into the very meaning of data, and real relations between aspects or variables. This all can help in supporting decision making, notably using the data driven decision support paradigm. The verbalization of data analyses results, in addition to the more commonly used visualization, e.g. via a GUI (graphical user interface), can contribute to an increased human consistency and ease of use because the results obtained (natural language sentences) are simple and easily comprehensible to the human being.

An important aspect of this paper was to present two new directions initiated by the authors. First, we mentioned the importance of a new concept of a *conceptual scalability* (cf. Kacprzyk and Zadrożny [1]), and showed the power of linguistic summaries in this respect. Second, maybe even more important, is a close relation of linguistic summaries to natural language generation (NLG) considered first by Kacprzyk and Zadrożny [2, 11], in which it was indicated that linguistic data summarization in the sense considered here is closely related to some types of solutions used in natural language generation (NLG), an area that is rapidly developing, and provides effective and efficient tools and techniques, and also ready to use software.

We are convinced that linguistic data summaries will play more and more relevant role in supporting human decision makers while solving difficult real life problems. And, more generally, verbalization may be an extremely relevant part of a human – computer interface in many applications, complementing and even sometimes replacing the commonly employed visualization.

REFERENCES

[1] J. Kacprzyk and S. Zadrożny, "Linguistic data summarization: a high scalability through the use of natural language?", *Scalable Fuzzy Algorithms for Data Management and Analysis: Methods and Design* 1, 214–237 (2009).

[2] J. Kacprzyk and S. Zadrożny, "Computing with words is an implementable paradigm: fuzzy queries, linguistic data summaries and natural language generation", *IEEE Trans. on Fuzzy Systems* 18, 461–472 (2010).

[3] S.L. Alter, *Decision Support Systems: Current Practice and Contributing Challenge*, Addison-Wesley, Reading, 1990.

[4] C.W. Holsapple and A.B. Whinston, *Decision Support Systems: a Knowldege-based Approach*, West Publishing, Minneapolis, 1986.

[5] R.H. Sprague and H.J. Watson, *Decision Support Systems for Management*, Prentice-Hall, Englewood Cliffs, 1996.

[6] L.A. Zadeh and J. Kacprzyk, *Computing with Words in Information/Intelligent Systems 1. Foundations. 2. Applications*, Physica-Verlag, New York, 1999.

[7] R.R. Yager, "A new approach to the summarization of data", *Information Sciences* 28, 69–86 (1982).

[8] J. Kacprzyk and R.R. Yager, "Linguistic summaries of data using fuzzy logic", *Int. J. General Systems* 30, 133–154 (2001).

[9] J. Kacprzyk, R.R. Yager, and S. Zadrożny, "A fuzzy logic based approach to linguistic summaries of databases", *Int. J. Applied Maths and Computer Science* 10, 813–834 (2000).

[10] J. Kacprzyk and S. Zadrożny, "Linguistic database summaries and their protoforms: towards natural language based knowledge discovery tools", *Information Sciences* 173 (4), 281–304 (2005).

[11] J. Kacprzyk and S. Zadrożny, "Protoforms of linguistic database summaries as a human consistent tool for using natural language in data mining", *Int. J. of Software Science and Computational Intelligence* 1 (1) 100–111 (2009).

[12] E. Reiter and R. Dale, *Building Natural Language Generation Systems*, Cambridge Univ. Press, Cambridge, 2000.

[13] J. Kacprzyk and P. Strykowski, "Linguitic summaries of sales data at a computer retailer: a case study", *Proc. IFSA'99* 1, 29–33 (1999).

[14] S. Zadrożny and J. Kacprzyk, "On database summarization using a fuzzy querying interface", *Proc. IFSA'99 World Congress* 1, 39–43 (1999).

[15] S. Zadrożny and J. Kacprzyk, "Summarizing the contents of Web server logs: a fuzzy linguistic approach", *Proc. IEEE-IS'2007* 1, 100–105 (2007).

[16] A.P. Wierzbicki, M. Makowski, and J. Wessels, *Model-based Decision Support Methodology with Environmental Applications*, Kluwer, Dordrecht, 2000.

[17] P.B. Checkland, *Soft Systems Methodology in Action*, Wiley, Chichester, 1990.

[18] P.B. Checkland, "Soft systems methodology. A thirty year retrospective", in *Soft Systems Methodology in Action*, ed. P.B. Checkland and J. Scholes, Wiley, Chichester, 1999.

[19] R.R. Yager, "Database discovery using fuzzy sets", *Int. J. of Intelligent Systems* 11, 691–712 (1996).

[20] P. Bosc, D. Dubois, O. Pivert, H. Prade, and M. de Calmes, "Fuzzy summarization of data using fuzzy cardinalities", *Proc. IPMU'2002* 1, 1553–1559 (2002).

[21] D. Dubois and H. Prade, "Fuzzy sets in data summaries-outline of a new approach", *Proc. 8th Int. Conf. IPMU* 2, 1035–1040 (2000).

[22] G. Raschia and N. Mouaddib, "SAINTETIQ: a fuzzy set-based approach to database summarization", *Fuzzy Sets and Systems* 129, 137–162 (2002).

[23] D. Rasmussen and R.R. Yager, "SummarySQL-a fuzzy tool for data mining", *Intelligent Data Analysis – An International Journal* 1, URL-http//:www-east.elsevier.com/ida/browse/96-6/ida96-6.htm (1997).

[24] D. Rasmussen and R.R. Yager, "Finding fuzzy and gradual functional dependencies with summary SQL", *Fuzzy Sets and Systems* 106, 131–142 (1999).

[25] J. Kacprzyk and S. Zadrożny, "On interactive linguistic summarization of databases via a fuzzy-logic-based querying add-on to microsoft access", *Computational Intelligence: Theory and Applications* 1, 462–472 (1999).

[26] J. Kacprzyk and S. Zadrożny, "On combining intelligent querying and data mining using fuzzy logic concepts", *Recent Research Issues on the Management of Fuzziness in Databases* 1, 67–81 (2000).

[27] J. Kacprzyk and S. Zadrożny, "Data mining via fuzzy querying over the Internet", *Knowledge Management in Fuzzy Databases* 1, 211–233 (2000).

[28] J. Kacprzyk and S. Zadrożny, "Data mining via linguistic summaries of databases: an interactive approach", *A New Paradigm of Knowledge Engineering by Soft Computing* 1, 325–345 (2001).

[29] J. Kacprzyk and S. Zadrożny, "Computing with words in intelligent database querying: standalone and Internet-based applications", *Information Sciences* 34, 71–109 (2001).

[30] J. Kacprzyk and S. Zadrożny, "Fuzzy querying for microsoft access", *Proc. FUZZ-IEEE'94* 1, 167–171 (1994).

[31] J. Kacprzyk and S. Zadrożny, "FQUERY for access: fuzzy querying for a Windows-based DBMS", *Fuzziness in Database Management Systems* 1, 415–433 (1995).

[32] T.H. Davenport and J.G. Harris, "What people want (and how to predict it)", *MIT Sloan Management Review* 50 (2), 22–31 (2009).

[33] R.A. Cole, J. Mariani, H. Uszkoreit, A. Zaenen, and V. Zue, *Survey of the State of the Art in Human Language Technology*, http://cslu.cse.ogi.edu/HLTsurvey/HLTsurvey.html, 1996.

[34] L.A. Zadeh, "A computational approach to fuzzy quantifiers in natural languages", *Computers and Maths with Appls.* 9, 149–184 (1983).

[35] R.R. Yager and J. Kacprzyk, *The Ordered Weighted Averaging Operators: Theory and Applications*, Kluwer, Boston, 1996.

[36] R.R. Yager, "On ordered weighted avaraging operators in multicriteria decision making", *IEEE Trans. on Systems, Man and Cybern*, SMC-18, 183–190 (1988).

[37] S. Zadrożny and J. Kacprzyk, "Issues in the practical use of the OWA operators in fuzzy querying", *J. Intelligent Information Systems* 33 (3), 307–325 (2009).

[38] R. George and R. Srikanth, "Data summarization using genetic algorithms and fuzzy logic", in *Genetic Algorithms and Soft Computing*, pp. 599–611, ed. F. Herrera and J.L. Verdegay, Physica-Verlag, Heidelberg, 1996.

[39] T.L. Saaty, *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*, McGraw-Hill, New York, 1980.

[40] L.A. Zadeh, "From search engines to question answering systems – the problems of world knowledge relevance deduction and precisiation", in *Fuzzy Logic and the Semantic Web*, pp. 163–210, ed. E. Sanchez, Elsevier, Amsterdam, 2006.

[41] J. Kacprzyk, S. Zadrożny, and A. Wilbik, "Linguistic summarization of some static and dynamic features of consensus

reaching", in: "*Computational Intelligence, Theory and Applications*", pp. 19–28, ed. B. Reusch, Springer, Berlin, 2006.

[42] F. Portet, E. Reiter, A. Gatt, J. Hunter, S. Sripada, Y. Freer, and C. Sykes, "Automatic generation of textual summaries from neonatal intensive care data", *Artificial Intelligence* 173, 789–816 (2009).

[43] S. Sripada, E. Reiter, and I. Davy, "SumTime-Mousam: configurable marine weather forecast generator", *Expert Update* 6 (3), 4–10 (2003).

[44] K.R. McKeown, *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*, Cambridge University Press, Cambridge, 1985.

[45] J. Kacprzyk and S. Zadrożny, "Soft computing and Web intelligence for supporting consensus reaching", *Soft Computing*, (2010), to be published.

[46] J. Kacprzyk and S. Zadrożny, "Towards a synergistic combination of Web-based and data-driven decision-support systems via linguistic data summaries", *LNAI* 3528, 211–217 (2005).

[47] J. Kacprzyk and S. Zadrożny, "On a concept of a consensus reaching process support system based on the use of soft computing and Web techniques", *Computational Intelligence in Decision and Control* 1, 859–864 (2008).