Towards bipolar linguistic summaries: a novel fuzzy bipolar querying based approach

Mateusz Dziedzic  
PhD Studies,  
Systems Research Institute  
Polish Academy of Sciences  
ul. Newelska 6, 01-447 Warszawa, Poland  
Email: Mateusz.Dziedzic@ibspan.waw.pl

Sławomir Zadrozny  
and Janusz Kacprzyk, IEEE Fellow  
Systems Research Institute  
Polish Academy of Sciences  
ul. Newelska 6, 01-447 Warszawa, Poland  
Email: {Slawomir.Zadrozny,Janusz.Kacprzyk}@ibspan.waw.pl

Abstract—we study the possibility to extend the concept of linguistic data summaries employing the notion of bipolarity. Yager’s linguistic summaries may be derived using a fuzzy linguistic querying interface. We look for a similar analogy between bipolar queries and the extended form of linguistic summaries. The general concept of bipolar query, and its special interpretation are recalled, which turns out to be applicable to accomplish our goal. Some preliminary results are presented and possible directions of further research are pointed out.

I. INTRODUCTION

Data querying and data mining may be seen as closely related but somehow opposite paradigms. The former is concerned with the retrieval of pieces of data satisfying the user’s interests, intentions and preferences while the latter is concerned with the finding of non-trivial, interesting to the user, regularities in data that are exhibited in general by the pieces of data. The results obtained may take very different forms in both cases. However, in a specific data mining technique of linguistic data summarization, introduced by Yager [1], [2], and in particular in its implementable version proposed by Kacprzyk and Yager [3], and Kacprzyk, Yager and Zadrozny [4], [5], there is an essential conceptual and algorithmic overlap in the structure of summaries and fuzzy linguistic queries which has been more formally and comprehensively exposed by Kacprzyk and Zadrozny [6]. This makes it possible to efficiently and effectively implement both functionalities in the framework of the same system, employing many common elements of the user interface.

What is characteristic to both linguistic data summaries and fuzzy linguistic queries is their use of the elements of natural language. The essence of the former is the use of natural language to describe characteristic features of a data set. The latter makes it possible for the user to express his or her preferences (conditions) as to the data sought using terms of natural language what is arguably often more convenient than using artificially and inadequately precise constraints, e.g. when looking for a cheap accommodation. In our previous work, cf. e.g. [6], [7], we were looking for the ways to exploit this close relationship between both techniques. In this framework we proposed an interactive approach to deriving linguistic summaries [8], [9].

Recently some attempts have been undertaken to make fuzzy linguistic queries even more human consistent in their representation of user preferences. Namely, the concept of bipolar queries has been introduced [10] in order to grasp the negative and positive assessments of data which are not necessarily complements of each other and expose strong attitudinal aspects. The bipolar queries may be considered in their most general form as consisting of two conditions separately accounting for these negative and positive assessments. However, in the literature a specific interpretation of bipolar queries seems to prevail so far in which the negative and positive conditions correspond to a constraint and a wish, respectively.

In this paper, we look for an extension of the concept of linguistic summaries which would exploit a relation to bipolar queries in the vein of a conceptually similar relation of linguistic summaries to fuzzy linguistic queries.

The structure of the paper is as follows. In the next section we briefly remind the essence of the fuzzy linguistic queries and linguistic summaries of data. In Section 3 we briefly discuss the concept of a bipolar query with an emphasis on its special interpretation. Section 4 comprises the main contribution of this paper introducing the concept(s) of bipolar linguistic summaries of data as well as pointing out some other possible uses of the bipolarity notion in this context. Finally, we conclude summarizing the content of the paper and envisaging the lines of possible further research.

II. FUZZY LINGUISTIC QUERIES AND LINGUISTIC SUMMARIES OF DATA

A. Fuzzy Linguistic Queries

While querying a database the user expresses his or her preferences as to the properties of the data sought. In the predominant relational data model this properties are basically expressed in terms of the attributes of the relations representing data. Usually these preferences are first expressed in natural language. Sometimes they are precise (as in, e.g., “give me the address of John Doe”) and may be easily translated to the standard languages of database querying, notably to SQL. However, quite often such a translation may be cumbersome, as in, e.g., “give me a list of cheap hotels in Brisbane” or in
“give me a list of seriously polluted parts of Copper Creek”. The difficulty with the former example lays in the vagueness of the term “cheap”. In the latter example “serious pollution” may require a non-standard – i.e., different from the simple conjunction or disjunction – aggregation of a number of some elementary conditions, e.g., related to exceeding threshold values of some basic indicators of pollution. Fuzzy logic has been very quickly found as to be a viable solution to address those difficulties. A lot of research has been carried out (and still is) addressing such issues as how to model linguistic terms such as “cheap”, “low” or “most”, and how to extend a query language so as to allow for the use of these linguistic terms. In the framework of fuzzy logic terms such as “cheap” are represented by fuzzy predicates while terms such as “most” are represented by linguistic quantifiers [11]. From the technical point of view, the fuzzy linguistic query interfaces are based on the dictionaries of such linguistic terms, cf., e.g., [12]. This way they provide an excellent environment for the implementation of the linguistic summaries of data which are briefly discussed in the next section.

B. Linguistic Summaries of Data

Linguistic summarization is a technique of data mining gaining recently more and more interest of the research community. Its underlying idea is to describe some interesting phenomena in large data sets – to summarize their essence – using very expressive, natural language expressions formally represented in the framework of fuzzy logic. There are many techniques available addressing seemingly a similar problem but they are not not human consistent, partly due to the fact that the use of natural language is limited, and this makes them less applicable in some scenarios. This concerns, e.g., summarizing statistics, exemplified by the average, median, minimum, maximum, α-percentile, etc. which – in spite of recent efforts to soften them – are still far from being able to reflect a real human perception of their essence.

The technique of linguistic data summarization has been originally proposed by Yager [11, 2] and further developed by many authors (cf., e.g., Kacprzyk and Yager [3], and Kacprzyk, Yager and Zadrożny [4, 5]). The essence of linguistic data summaries is that a set of data, say, concerning employees, with (numeric) data on their age, sex, salaries, etc., can be summarized linguistically with respect to a selected attribute or attributes, say age and salaries, by linguistically quantified propositions [11] such as

\[
\text{Almost all employees are well qualified} \quad (1)
\]

or

\[
\text{Most young employees earn high salary} \quad (2)
\]

The linguistic summaries may be considered in the following context:

- \( R = \{ t_1, \ldots, t_n \} \) is a set of the tuples (a relation \( R \)) in a database, representing, e.g., a set of employees;
- \( A = \{ A_1, \ldots, A_m \} \) is a set of attributes characterizing relation \( R \), e.g., salary, age, etc. in a database of employees, and \( A_j(t_i) \) denotes a value of attribute \( A_j \) for tuple \( t_i \).

Then, a linguistic summary of set \( R \) is a linguistically quantified proposition such as (1) or (2) which may be treated as instantiations of an abstract protoforms [13], respectively:

\[
Q_{t \in R} \ S(t) \quad (3)
\]

and

\[
Q_{t \in R} \ (U(t), S(t)) \quad (4)
\]

where the predicates \( S \) and \( U \) correspond to conditions formed using attributes of the set \( A \). The following parts of a linguistic summary are distinguished:

- a summarizer \( S \) is a fuzzy predicate representing, e.g., an expression “an employee is well qualified”;
- optionally, a qualifier \( U \) which is another fuzzy predicate determining a fuzzy subset of \( R \) (e.g. a set of “young employees”) to which the summarizer is meant to apply;
- a linguistic quantifier \( Q \), e.g., “most”, representing the proportion of tuples satisfying the summarizer (optionally, among those satisfying a qualifier);
- truth (validity) \( T \) of the summary, i.e. a number from the interval \([0, 1]\) assessing the truth (validity) of the summary (e.g. 0.7); usually, only summaries with a high value of \( T \) are interesting.

The truth value \( T \) may be determined according to many different models of linguistic quantifiers, notably one originally proposed by Zadeh [11]. Using Zadeh’s fuzzy calculus of linguistically quantified propositions, a (proportional, non-decreasing) linguistic quantifier \( Q \) is assumed to be a fuzzy set in \([0, 1]\) and then

\[
\text{truth}(Q_{t \in R} S(t)) = \mu_Q \left[ \frac{1}{n} \sum_{i=1}^{n} \mu_S(t_i) \right] \quad (5)
\]

and

\[
\text{truth}(Q_{t \in R} (U(t), S(t))) = \mu_Q \left[ \frac{\sum_{i=1}^{n} (\mu_U(t_i) \land \mu_S(t_i))}{\sum_{i=1}^{n} \mu_U(t_i)} \right] \quad (6)
\]

Protoforms [13] are a more or less abstract prototypes (templates) of a linguistically quantified proposition. The most abstract protoforms correspond to (3) and (4), while (1) and (2) are examples of fully instantiated protoforms. Thus, protoforms form a hierarchy, where higher/lower levels correspond to more/less abstract protoforms. Going down this hierarchy one has to instantiate particular components of (3) and (4), i.e., \( Q \), and \( S \) and \( U \). A protoform may provide a guiding paradigm for a user interface for the mining of linguistic summaries. In our interactive approach to linguistic summaries mining [8, 9] it is assumed that the user provides the system with a protoform of a summary. The system is meant as an extension of a fuzzy linguistic querying interface and is equipped with a dictionary of linguistic terms, mentioned in section II-A. Then, the system generates and tests linguistic
summaries in the context of a given database which result in various instantiations of the protoform provided, using different combinations of linguistic terms from the dictionary.

The user is presented with those summaries which verify the validity to a degree higher than some threshold. The more abstract protoform the more instantiations are possible and the resulting summaries are potentially more interesting while the computation costs are higher. This high computational costs may be somehow alleviated by the use of data statistics gathered by database management systems which may help in choosing first most promising instantiations of the protoforms in terms of the validity of resulting summaries.

The linguistic summaries, such as (1)–(2), are easily comprehensible for a human being in their “raw” form and may be in terms of the validity of resulting summaries.

choosing first most promising instantiations of the protoforms gathered by database management systems what may help in the validity to a degree higher than some threshold. The more different combinations of linguistic terms from the dictionary.

in various instantiations of the protoform provided, using summaries in the context of a given database which result in various instantiations of the protoform provided, using different combinations of linguistic terms from the dictionary.

In this paper we consider a special class of bipolar queries where the positive and negative conditions are interpreted in a specific way (for an approach to bipolar queries meant in the general form mentioned above cf., e.g., [16]). Namely, the data items sought have to satisfy the complement of the latter conditions unconditionally, while the former conditions are of a somehow secondary importance. For example, a hotel the user is looking for may have to be cheap, and then among such houses those which are comfortable are preferred. The negative condition is here “not cheap” while the positive condition is “comfortable”. It is then natural to specify the complement of the negative condition in such a query (denoted \( C \)), which may therefore be interpreted as a required condition. On the other hand, the positive condition is expressed directly and may be referred to as a desired condition (denoted \( P \)), and the whole query will be denoted \((C, P)\). Both conditions are generally fuzzy and may be identified with fuzzy sets, the membership functions of which will be denoted \( \mu_C \) and \( \mu_P \), respectively.

The evaluation of a bipolar query results in two membership (matching, satisfaction, ...) degrees \((\mu_C(t), \mu_P(t))\), computed for each tuple \( t \). In order to make the concept of bipolar queries operational we have to adopt some method to compare such pairs of membership degrees. Following Lacroix and Lavency [17], Yager [18], [19] and Bordogna and Pasi [20], we model the interaction between the required and desired conditions using the “and possibly” operator casting the whole bipolar query condition in the following form:

\[
C \text{ and possibly } P
\]  

which may be exemplified, referring to the previous example, by:

\[
\text{cheap and possibly comfortable}
\]

The essence of the “and possibly” operator consists in taking into account the whole dataset while combining the matching degrees related to the required and desired conditions. Yager [19] refers to \( P \) as the possibilistically qualified criterion which is intuitively characterized as one which should be satisfied unless it interferes with the satisfaction of \( C \). This is in fact the essence of the aggregation operator “and possibly” as we understand it here. Namely, it is assumed that if there is a piece of data which satisfies both conditions, then and only then it is actually possible to satisfy both of them and each piece of data has to meet both of them. Thus, the \((C, P)\) query reduces to the usual conjunction \( C \land P \). On the other hand, if there is no such a piece of data, then it is not possible to satisfy both conditions and the desired one can be disregarded. Thus, the \((C, P)\) query reduces to \( C \). These are however two extreme cases and actually it may be the case that the two conditions may be simultaneously satisfied to some degree which is however less than 1. Then, the matching degree of the \((C, P)\) query against a piece of data lies somewhere between its matching degrees of \( C \land P \) and \( C \). This may be formally written for the crisp case as [17]:

\[
C(t) \text{ and possibly } P(t) \equiv C(t) \land (\exists s(C(s) \land P(s)) \Rightarrow P(t))
\]

and for the fuzzy case as [21], [22]:

\[
\text{truth}(C(t) \text{ and possibly } P(t)) = \min(\mu_C(t), \max(1 - \max \min(\mu_C(s), \mu_P(s)), \mu_P(t)))
\]

where \( R \) denotes the whole dataset (relation) being queried. The value of

\[
\max \min(\mu_C(s), \mu_P(s)), \text{ denoted as } \exists CP
\]

which expresses the truth of \( \exists s(C(s) \land P(s)) \), may be treated as a measure of the above mentioned interference of \( P \) with \( C \).

The formula (11) is derived from (9) using the classic fuzzy interpretation of the logical connectives via the maximum and minimum operators. In Zadrozny and Kacprzyk [22] we analyze the properties of the counterparts of (10) obtained by using a broader class of the operators modeling logical connectives.
IV. Bipolarity in Linguistic Summaries: Why and How

In what follows we assume that a system is available that makes it possible to execute fuzzy linguistic queries and generate linguistic summaries—such as our FQUERY for Access package [12], and such a system has been extended with the capability to execute bipolar queries in the sense of the required/desired semantics described in the previous section. Now we will consider some possible interpretations of bipolar linguistic summaries and how they may be generated in such an environment.

Let us start with an observation that the straightforward use of the formula (7) to instantiate the summarizer in the linguistic summary protoforms (3)–(4) and its interpretation via (8) does not make much sense. Namely, the expression (7) may be appropriate to represent preferences of the user (as it is exemplified by (9)) who does not know if there is an interference between conditions C and P with respect to the content of a database queried. However, e.g., the following proposition:

Most of the employees are young and possibly earn high salaries.

is rather meaningless in the role of a summary as the system knows if there is or not such an interference. However, we will show that knowledge of such an interference may be interesting on its own for the user in some cases and, moreover, that some slight modification to (9) may give rise to some forms of linguistic summaries involving “and possibly” operator and still worth study and preserving the bipolar flavor.

A. Simple Bipolarity-Related Summaries

Thus in what follows we assume an interactive approach to the generation of linguistic summaries. As mentioned in Section II-B, such an approach may benefit from statistics gathered by the database management systems. In case of bipolar queries execution those statistics may easily include the \( \exists C P \) coefficient (11). This way, the system may immediately generate a summary of the form:

It is possible to satisfy simultaneously conditions \( C \) and \( P \) which may be phrased by the postulated use of the NLG techniques (cf. section II-B) of the system in a slightly different way depending on the semantics of the predicates \( C \) and \( P \), what may be exemplified for the previously considered personnel database as follows:

There are young employees earning a high salary.

Thus, it motivates the instantiation of the quantifier \( Q \) in the protoform of linguistic summaries \( Q \in R S(t) \) (as well as in \( Q \in R (U(t), S(t)) \) also with the existential quantifier \( \exists \) – here \( Q \) and \( S \) are instantiated with \( \exists \) and \( C(t) \land P(t) \), respectively. This makes sense especially in the case of somehow semantically “conflicting” predicates \( C \) and \( P \). It should be observed that the lack of such a summary among those generated by the system (due to its low validity degree) may also be very informative as it actually shows this conflicting nature of two predicates in the context of a database under consideration what may be not obvious for the user. Finally, it should be noted that a linguistic summary \( Q \in R C(t) \land P(t) \) - with the linguistic quantifier \( Q \) of the “most” type makes \( \exists \in R C(t) \land P(t) \) redundant. Thus, the system should consider generating the latter only in case the former has not been generated (i.e., its validity measure is not high enough).

Even if such a summary does not refer directly in its construction to the bipolarity it should be noted that its interestingness becomes apparent from our considerations of bipolar queries (cf. Section IV).

B. Bipolar Linguistic Summaries with a Local “and possibly” Operator

An interpretation of the proposition “\( C \) and possibly \( P \)” (7) given by (9) makes this proposition true (to a high degree) for a tuple \( t \) only if either of two conditions holds:

1) it satisfies (to a high degree) both conditions \( C \) and \( P \), or

2) it satisfies \( C \) and there is no tuple in the whole database which satisfies both conditions.

We noticed at the beginning of this section that such an interpretation does not make the proposition “\( C \) and possibly \( P \)” a good candidate for a summarizer. Now we will introduce another version of the “and possibly” operator with an interpretation weakening conditions (12), notably the second one. Namely, let us relate the notion of “possibility”, present in the “and possibly” interpretation, to a part of the database rather than the whole database and, moreover, let us make this part dependent on the tuple which is considered. Thus, in order to decide if simultaneous satisfaction of both conditions \( C \) and \( P \) is possible we will now check only a subset of the tuples and this subset will be defined locally, separately for each tuple. Before formalizing this idea, let us motivate it with an example. Let us take into account that a company is divided into departments and each employee is assigned to one of them. Then, in the linguistic summary:

Most employees are young and possibly with respect to their department mates earn high salary.

the summarizer “young and possibly with respect to their department mates earns high salary” should be meant satisfied by an employee if:

1) he or she is young (to a high degree) and earns a high salary (to a high degree), or

2) he or she is young (to a high degree) and there is no other employee in his or her department who is both young and earns a high salary.

Such a summary is now meaningful as it provides the user with a potentially interesting piece of information.

Let us formalize this new version of the “and possibly” operator which we will refer to in what follows as the “local
and possibly operator”. It is no more a binary operator but it admits three arguments:

\[ C \text{ and possibly } P \text{ with respect to } W \]  

where predicates \( C \) and \( P \) should be interpreted, as previously, as representing the required and desired conditions, respectively, while the predicate \( W \) denotes the set of tuples.

Referring to our example, \( C \) and \( P \) represent the properties of being young and earning high salary, respectively, while \( W \) denotes the relation of working in the same department, i.e., \( W(t, s) \) is true if both tuples represent employees working in the same department. In this case \( W \) is a crisp predicate but in general it may be a fuzzy one. For example, we could modify example summary (13) in the following way:

Most employees are young and possibly with respect to similarly educated colleagues earn high salary.

and then \( W \) may be a fuzzy relation based on the attribute representing the level of education of the employees and expressing to which degree they are similar (e.g., M.Sc. may be treated as similar to B.Sc. to a degree 0.6 etc., depending on the context).

It should be stressed that we follow here an interactive approach to linguistic summarization which seems to be the most promising, in the case of bipolar linguistic summaries too. Thus, the choice of \( C \), \( P \) and \( W \) is up to the user who is specifying a protoform that is proper for the problem concerned, but also to the system which is instantiating it in possibly efficient and effective way. However, it seems beneficial to have the predicates \( W \) predefined, at least to some extent. This is not a serious detour from the case of earlier considered linguistic summaries for which a dictionary of potential summarizers, qualifiers and linguistic quantifiers has also to be available.

### C. Linguistic Summaries with a Grouping Based “and possibly” Operator

The linguistic summaries originally proposed by Yager [1], and Kacprzyk and Yager [3], and Kacprzyk, Yager and Zadrozny [4, 5], as well as their extended bipolar form proposed in the previous subsection concern individual tuples. In other words, they typically identify properties shared by some majority of tuples in the whole database or in a (fuzzy) subset of tuples. However, it may be even more interesting to form summaries concerning the sets of tuples. Let us consider the following examples:

Most departments employ young and earning high salaries employees. (17)

or

In almost all Australian cities most of the hotels are moderately priced and comfortable. (18)

Thus, we have here two linguistic quantifiers: the “outer” concerning the groups of the tuples and the “inner” related to the tuples in a group. In the former example such an inner quantifier is implied and corresponds to the general quantifier “for all”.

Before we show how bipolarity may be introduced in this new type of summaries, it is worthwhile to confront them with a still another type of summaries exemplified by our summaries of time series as proposed by Kacprzyk, Wilbik and Zadrozny [23]. In this approach database tuples, representing a time series data, are also grouped and only then summarized. The difference is however that each group of tuples, corresponding to a trend in the data set, is treated as a new individual tuple characterized by some properties of the whole group of original tuples (e.g., the dynamics of the trend). Thus, the summaries generated follow the general protoform (5) or (4) with some slight variation related to its specific interpretation. In the approach proposed here we introduce a new protoform which may be formally expressed as:

\[ Q^{1}_{y}Q^{2}_{x \in y}S(x) \]  

In the examples shown earlier this protoform is instantiated as follows:

for (17)

\[ Q^{1} = "most", \quad Q^{2} = "all", \quad Y = \{y_{i}\} \text{ is a set of departments}, \quad X_{y} = \{x_{i}\} \text{ is a set of employees of a department } y, \quad \text{and } S(x) = \{x \text{ is young and highly paid (earns high salary)}\}, \]

for (18)

\[ Q^{1} = "almost all", \quad Q^{2} = "most", \quad Y = \{y_{i}\} \text{ is a set of cities in Australia}, \quad X_{y} = \{x_{i}\} \text{ is a set of hotels located in a city } y, \quad \text{and } S(x) = \{x \text{ is moderately priced and comfortable}\}. \]

The truth degree of a summary represented by the protoform (19) is computed using a modified version of formula (5) used for standard linguistic summaries prototypes (5):

\[
\text{truth}(Q^{1}_{y}Q^{2}_{x \in y}S(x)) = \min \left( \frac{\sum_{i=1}^{\left| y_{1} \right|} \mu_{Q^{1}} \left( \sum_{j=1}^{\left| y_{i} \right|} \mu_{Q^{2}}(x_{j}) \right)}{\| y \|} \right)
\]

where \( |A| \) denotes the cardinality of set \( A \).

Let us observe that in the case of the new protoform (19) the use of the “and possibly” connective becomes reasonable and interesting. Let us consider the following modification of the example of summary (17):

Most departments employ young and possibly earning highly paid employees. (21)
which should be understood as follows: it is true (to a high degree) for most departments that they employ only young people and, moreover, if such a department employs a young and earning a high salary employee (i.e., it possible to be young and earn a lot in this department) then all employees in it are young and earn a high salary.

Thus, if a summarizer in (19) contains the “and possibly” connective then the notion of possibility is interpreted in the context of a set $y$ rather than in the context of the whole database. Formally, this is expressed for a linguistic summary:

$$Q_y^1Q_y^2 C(x) \text{ and possibly } P(x)$$  \hspace{1cm} (22)

by the following form of $\mu_S(x_j)$ in the formula (20) for its truth value:

$$\mu_S(x_j) = \text{truth}(C(x_j) \land \exists x_k \in y, (C(x_k) \land P(x_k)) \rightarrow P(x_j))$$  \hspace{1cm} (23)

In the next section we discuss some aspects of thus introduced bipolar linguistic summaries of data.

V. DISCUSSION AND RELATED WORK

Some similarity between the bipolar linguistic summaries with local “and possibly” operator and bipolar linguistic summaries with the grouping based “and possibly” operator is worth noting and discussing. Namely, both types of summaries are characterized by the interpretation of the “and possibly” operator in a restricted context: the possibility of the satisfaction of both arguments of this operator is defined in terms of a subset of tuples rather than the whole database. In the former case a relation $W$ is introduced to form the context - in general, separately for each tuple. In the latter case a partitioning $Y = \{y_i\}$ of the set of tuples is assumed which is fixed for the whole database. Both, relation $W$ and partitioning $Y$, may be fuzzy what makes the modeling of the context more realistic and slightly more complicated computationally, at the same time.

Let us now consider, for the sake of simplicity, the case of the crisp: relation $W$ and partitioning $Y$. Then, the latter case seems to be a special case of the former for $W$ being an equivalence relation. However, we should notice that the protoform (5) with the summarizer (15) is essentially different from the protoform (12). The latter makes use of two level linguistic quantifier guided aggregation and concerns groups of tuples while the former concerns individual tuples. Thus even apparently very similar summaries of these two types convey different information and should in general have different truth degrees. This may be observed comparing two summaries (13) and (21). Let there are 5 departments employing the numbers of young workers shown in Table I. For the sake of simplicity we assume that the predicates “young” and “earns high salary” are both crisp. Then, the summary (13) will have a low truth degree as even if most of the employees are young (90 out of 100) still for almost a half of them it is not true that they earn high salary while they have young department mates who do (cf. department I). On the other hand, in most of the departments (4 out of 5) there are working only young employees and in each of these departments they all earn high salaries or no one of them earns high salary, thus the summary (21) will have a high truth degree.

<table>
<thead>
<tr>
<th>Dept. no.</th>
<th>Total no. of employees</th>
<th>No. of young employees</th>
<th>No. of young employees earning high salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>60</td>
<td>50</td>
<td>2</td>
</tr>
<tr>
<td>II</td>
<td>10</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>III</td>
<td>10</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>IV</td>
<td>10</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>V</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

This comparison well illustrates a need for new measures of quality in case of bipolar linguistic summaries. Namely, the validity $T$ corresponding to the truth of a summary, (5) or (6), is the basic measure of the quality for regular linguistic summaries. There are several other quality measures proposed, cf., e.g., [4], [5] which take into account different aspects of the summary. In case of bipolar linguistic summaries still another measure is needed which, roughly speaking, takes into account how often the summarizer “$C$ and possibly $P$” operator involved turns into “$C \land P$” or into “$C$”; cf. Section 11-B. Namely, if it happens for a high majority of tuples (in case of a linguistic summary with local “and possibly” operator) or for a high majority of groups of tuples (in case of a grouping based “and possibly” operator), then the bipolar summarizer may be more often satisfied than its corresponding “standard summarizers” (“$C \land P$” or “$C$”) while the resulting summary is less interesting than the standard one. A basic component of such a measure is the $\exists CP$ coefficient which should always be taken into account when deciding the quality measure of a bipolar linguistic summary.

Bipolar linguistic summaries with the grouping based “and possibly” operator are easier to integrate with the standard SQL queries, and thus also with their fuzzy linguistic extensions. Namely, the partitioning $Y$ may be defined by the GROUP BY clause of the SELECT command. In case of a fuzzy partitioning things get more complex but the general idea is still in line with the well-known syntax of the SQL and related intuitions of the users of this language.

Concerning a related research we are not aware of any other extensions of the concept of linguistic summaries which take into account the bipolarity of the constituent conditions, in the sense assumed here. Maybe, the work of Niewiadomski [24] should be mentioned here. Niewiadomski introduces the concept of Type-2 linguistic summary where the all linguistic terms in a summary, i.e., linguistic quantifiers and fuzzy predicates in summarizers and qualifiers, may be in general represented using Type-2 fuzzy sets instead of “regular” fuzzy sets. Some links with bipolar linguistic summaries proposed here by us become evident especially when compared with interval-valued fuzzy sets based summaries which are proposed by Niewiadomski as a special case of Type-2 fuzzy sets. Interval-valued fuzzy sets may be in turn interpreted in
terms of Atanassov intuitionistic fuzzy (A-IFS) sets which are one of the possible theoretical foundations of the bipolar queries; cf., e.g., [13, 25]. However, there is a fundamental difference between our approach and that of Niewiadomski. Namely, in the latter the interval-valued fuzzy sets are used to model individual linguistic terms in the domains of particular attributes. When interpreted in terms of A-IFS it makes it possible to, e.g., indicate the ranges of prices which are to some extent cheap (positive, desired) and not cheap (negative, complement of required), separately - not requiring that one range is the complement of another. In our approach bipolarity of a summarizer or a qualifier may be more subtle: the definitions of desired and required components may refer to different attributes what seems to lead to more interesting summaries in some practical scenarios. The links between both approaches surely deserve a further study and the difference indicated above is related to the structural aspects of bipolarity expression discussed in our earlier paper [26].

VI. CONCLUSION

In this paper we proposed a novel concept of a bipolar linguistic summary with the “and possibly” operator. It makes it possible to discover some regularities present in the data which are not visible when the possible bipolarity is neglected. Moreover, the resulting summaries should be fairly easily understandable for the user as it is in the case of bipolar queries.

Due to a conceptual and foundational character of the paper, for the sake of clarity we have proposed here only qualifier-free versions of the bipolar linguistic summaries. They may be extended to include qualifiers in an obvious way. Some possible and less evident interpretations of bipolar linguistic queries with qualifiers will be a subject of further studies.

In terms of a possible efficient and effective implementation of the mining of such bipolar summaries we propose to exploit the analogy to our earlier approach to derive regular linguistic summaries through a human-computer graphical interface to execute fuzzy linguistic queries.

Further research is needed to study some aspects of the proposed approach. For example, the protoform (19) should be studied from the perspective of higher order constructs which are used therein. Another perspectives which may be adopted to analyze the approach proposed is based on modal logic, as proposed by Kacprzyk and Zadrozny [27] or using Grabisch, Greco and Pirlot’s [28] bipolarity based MCDA model.

ACKNOWLEDGMENT

Mateusz Dziedzic contribution is supported by the Foundation for Polish Science under International PhD Projects in Intelligent Computing. Project financed from The European Union within the Innovative Economy Operational Programme (2007-2013) and European Regional Development Fund.

REFERENCES


