News Generating Based on Interval Type-2 Linguistic Summaries of Databases

Adam Niewiadomski
1 Institute of Computer Science, Technical University of Łódź, Poland
aniewiadomski@ics.p.lodz.pl

Piotr S. Szczepaniak
1 Institute of Computer Science, Technical University of Łódź, Poland
2 Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland
piotr@ics.p.lodz.pl

Abstract

Generating the so-called linguistic summaries of databases, exemplified by About half of records have very high values of attribute A, in sense of Yager [17] [19] with further improvements [2] [5] is described. The original authors’ contribution is the use of Interval Type-2 Fuzzy Sets [8] [9], instead of ordinary fuzzy sets, to model imprecise but understandable natural language propositions like high cost, small amount. It is necessary to obtain the knowledge representation which is not arbitrary, but build on opinions given by different experts and/or collected from different sources. The described methods provide the foundations for automated generating of textual news to be published in e-press, WWW, or in RSS channels. A sample implementation and its results are presented.

Keywords: Linguistic summaries of databases, knowledge mining, information summarization, fuzzy sets, type-2 fuzzy sets, interval type-2 fuzzy sets.

1 Introduction and motivation

It is almost trivial to say that the contemporary society is dependent on information. Nevertheless, a serious problem appears when all data needed to control various processes, apart from being found and retrieved, must also be reproduced in an easy-to-use and human-consistent form. In many circumstances, a communicative and compact but not necessarily precise message is required to be obtained quickly rather than by a thorough but long-lasting analysis. Therefore, the use of methods which present data in natural language (NL), should be considered. Interesting approaches of reporting large datasets via traditional fuzzy sets are given by Kacprzyk and Zadrozny [6] [7].

In this paper, the idea of linguistic summarization of databases is originally adapted to generate NL messages which are used in formulating press comments, memos, news, etc. Moreover, the approach is originally enriched by the usage of Interval Type-2 Fuzzy Sets (IT2FS).

2 Fuzzy sets

2.1 The Zadeh approach

In the classic set theory, a set A in a universe X can be represented by its characteristic function \( \chi_A: X \to \{0,1\} \) which shows whether an element belongs to the set or not, and tertium non datur. Nevertheless, it frequently happens in NL that the so-called partial belongingness of an element to a set must be considered. The concept of a fuzzy set represented by its membership function (MF) was introduced by Zadeh in 1965; it extends the set of values of a characteristic function to the
[0, 1] interval [20]:

$$A = \{ \langle x, \mu_A(x) \rangle : x \in X \}$$  \hspace{1cm} (1)

where $\mu_A : X \rightarrow [0, 1]$ is the membership function of the A fuzzy set; its value for $x \in X$ is interpreted as the membership level of $x$ to $A$. The generalization enables representing linguistic statements like old man, very fast car, which are inherently imprecise, but understandable in NL.

### 2.2 Type-2 fuzzy sets

The main idea of a Type-2 Fuzzy Set is that each type-2 membership degree is a type-1 fuzzy set assigned to an element of $X$. It allows one to construct imprecise or incomplete membership levels, since an ordinary MF with real values is said to be too terse to represent uncertainty [8, 10]. Formally, a T2FS $\tilde{A}$ in $X$ is

$$\tilde{A} = \{ \langle x, u, \mu_x(u) \rangle : x \in X, u \in J_x \}$$  \hspace{1cm} (2)

where $u$ is the primary membership degree of $x$, and $\mu_x(u)$ is the secondary membership level, specific for a given pair $\langle x, u \rangle$, $u \in J_x \subseteq [0, 1]$. Different kinds of T2FSs can be described, and they are determined only by secondary MFs; one may consider Gaussian T2FSs, triangular, interval, etc. In particular, we are interested in applying Interval T2FSs in this approach. Their semantics is based on the fact that technical and natural data are frequently expressed by intervals rather than by crisp numbers. Similarly, membership levels may also be approximated by intervals, e.g. $[0.7, 0.8]$, whenever an expert hesitates or his knowledge is estimated by some ranges of possible values etc. Therefore, two MFs are considered in an IT2FS: lower (LMF) and upper (UMF). Hence, an IT2FS $\tilde{A}$ in $X$ may be represented as:

$$\tilde{A} = \{ \langle x, \mu_\tilde{A}(x), \overline{\mu}_\tilde{A}(x) \rangle : x \in X \}$$  \hspace{1cm} (3)

where $\mu_\tilde{A}, \overline{\mu}_\tilde{A} : X \rightarrow [0, 1]$ are the lower MF and the upper MF, respectively. The secondary membership function does not specify $\tilde{A}$ since:

$$\mu_x(u) = \begin{cases} 1, & \text{if } \mu_\tilde{A}(x) \leq u \leq \overline{\mu}_\tilde{A}(x) \\ 0, & \text{otherwise} \end{cases}$$  \hspace{1cm} (4)

Hence, it is rectangular. Notice that in this case each membership level is an interval in $[0, 1]$. Notice also that if $\forall x \in X \mu_\tilde{A}(x) = \overline{\mu}_\tilde{A}(x)$, then $\tilde{A}$ can be considered as the ordinary fuzzy set, which is crucial for generalizing the original approach to linguistic summarization.

### 2.3 Linguistic variables

The concept of a linguistic variable (LV) is based on fuzzy sets. It comes directly from Zadeh, who explains that The concept of a linguistic variable provides a means of approximate characterization of phenomena which are too complex or too ill-defined to be amenable to description in conventional quantitative terms... [21].

The main reason for extending the idea of LV is that ordinary fuzzy sets describe imprecise information with crisp numbers, which may be seen contradictory. Moreover, in many cases, membership degrees are described by people with words again, instead of crisp numbers. Hence, the T2FS-based extension of LV is defined:

**Definition 1** A type-2 linguistic variable (T2LV) is an ordered quintuple $(L, H, X, G, M)$, where $L$ is the name of the variable, $H$ or $H(L)$ is the term-set of linguistic values of $L$, $X$ is the universe of discourse, $G$ is a syntactic rule which generates the terms (labels) in $L$, $M$ is a semantic rule which associates a term from $L$ with a T2FS in $X$.

The definition is a generalization of the Zadeh definition, since a type-2 set is a generalization of a type-1 set. In particular, in linguistic summarization, at least three possibilities of employing T2LVs may be enumerated: 1) as a summarizer, 2) as a linguistic quantifier (in this case $X \subseteq \mathbb{R}^+ \cup \{0\}$), and 3) as a query $w_g$. The idea of 1) and 2) cases has already been introduced in [11]. Nevertheless, this paper additionally provides technical details and a real application.

The model of the AND connective for two or more labels associated with a T2FS each, can

---

1A similar idea has been presented by Mendel [10].
be found via the meet operation on their MFs, which produces the intersection for T2FSs [8]:

$$\tilde{A} \cap \tilde{B} \leftrightarrow \int_{u_A} \int_{u_B} \frac{\mu_A(x, u_A) \cdot t_1 \cdot \mu_B(x, u_B)}{u_A \cdot t_2 \cdot u_B}$$  \hspace{1cm} (5)

where $u_{\tilde{A}}$, $u_{\tilde{B}}$ are primary membership degrees for $x$ in $\tilde{A}$, $\tilde{B}$, respectively, and $t_1$, $t_2$ - $t$-norms (in a discrete $X$, integrals can be replaced by summations). The special case of meet for IT2FSs is computed as a $t$-norm for values of LMF and UMF, respectively, and the obtained result is an interval which is interpreted as the compatibility level of a statement.

2.4 Linguistic quantifiers

NL statements which enable grasping amounts of collected individuals manifesting attributes of interest, are called natural language quantifiers, e.g. most, very few, from one to three, etc. The most known method of modelling quantifiers is the original Zadeh’s fuzzy quantification [22], and we follow this approach by presenting linguistic summaries of database. Two kinds of Zadeh’s fuzzy quantifiers can be distinguished: absolute which is a fuzzy set in any positive universe of discourse, e.g. over 190, about 1000, less than 10, and relative which is a fuzzy set in [0, 1] and expresses an amount of interest as ratio to the cardinality of the whole universe; e.g. about 1/4, almost all, few. Both kinds are usually represented as fuzzy sets on $\mathbb{R}^+ \cup \{0\}$ and on [0, 1], respectively.

3 Linguistic data summaries

3.1 Generating summaries

In order to obtain compact and communicative (though not strictly precise) information about the content of a given database, natural language sentences which describe amounts of elements manifesting properties, can be generated. They are called linguistic summaries of databases. The basic Yager approach to linguistic summaries [17] [18], and its further extensions [2] [4] are presented in this section.

**Yager’s summaries (Y)** Let us define the set of objects $\mathcal{Y} = \{y_1, ..., y_m\}$, the set of attributes $V = \{V_1, ..., V_n\}$. Let $X_1, ..., X_n$ be the domains of $V_1, ..., V_n$, respectively. Attributes from $V$ describe objects from $\mathcal{Y}$; it is denoted as $V_j(y_i)$ — a value of attribute $V_j$ for object $y_i$. Hence, a database $D$ which collects information about $\mathcal{Y}$, is

$$D = \langle V_1(y_1), ..., V_n(y_1), \cdots, V_1(y_m), ..., V_n(y_m) \rangle = \{d_1, ..., d_m\}$$  \hspace{1cm} (6)

where $d_1, ..., d_m$ are the records which describe objects $y_1, ..., y_m$, respectively, such that $d_i \in X_1 \times ... \times X_n$. $S_1, ..., S_n$ are the labels represented by fuzzy sets in $X_1, ..., X_n$, respectively. Let $Q$ be a fuzzy quantifier. A linguistic summary is in the form of

$$Q \ P \ are/\ have \ S_j \ [T]$$  \hspace{1cm} (7)

where

$$T = \mu_Q \left( \frac{\sum_{i=1}^m \mu_{S_j}(V_j(y_i))}{M} \right)$$  \hspace{1cm} (8)

where $M = m$ if $Q$ is relative, or $M = 1$ if $Q$ is absolute. $P$ is the subject of summary, and $S_j$ is the summarizer. $T \in [0, 1]$ is a degree of truth of the summary and is interpreted as a quality measure of a summary: the closer to 1 it is, the more reliable the summary is.

**George and Srikanth’s summaries (G)** It is worth adding that only summaries built of a single summarizer are considered in the original Yager approach i.e. $S = S_i, i = 1, ..., n$. George and Srikanth [2] proposed building summarizers of a few fuzzy sets using a $t$-norm (originally, the minimum):

$$\mu_S(d_i) = \min_{j=1,2,...,n} \{ \mu_{S_j}(V_j(y_i)) \}$$  \hspace{1cm} (9)

and the formula for $T$ is still (8) in which $S_j := S$. A sample (G) summary is: *Almost none of my friends is young and reach.*

**Kacprzyk, Yager, and Zadrozny’s summaries (K)** Kacprzyk, Yager, and Zadrozny [5] proposed generating summaries
in which one of summarizers is chosen as the so-called query (denoted as \( w_q \)):

\[
Q \ P \text{ are/have} \ S \ [T] \quad (10)
\]

The form of \( \mu_S \) is:

\[
\mu_S(d_i) = \min_{j=1,\ldots,n} \left\{ \mu_{S_j}(V_j(y_i)) \cdot \mu_{w_q}(V_q(y_i)) \right\} \quad (11)
\]

where the cofactor \( \mu_{w_q}(V_q(y_i)) \) means that only the tuples with non-zero membership degrees to the feature \( q \) — \( \mu_{w_q}(V_q(y_i)) > 0 \) — are considered in the final result; other records are not. The \( T \) index is:

\[
T = \mu_Q \left( \frac{\sum_{i=1}^{m} \mu_{S}(d_i)}{\sum_{i=1}^{m} \mu_{w_q}(V_q(y_i))} \right) \quad (12)
\]

(for relative quantifiers only). The mechanism, apart from the meaningful cost reduction, provides also more specific and informative summaries.

### 3.2 Interval Type-2 Linguistic Summaries

We introduce the use of type-2 linguistic variables in modeling of features of objects in databases. The analogous idea, but based on interval-valued fuzzy sets has already been presented in [12], [13], [15]. A linguistic variable, the values of which are associated with IT2FSs, allows one to operate on intervals as on compatibility levels. Here, we use interval type-2 fuzzy sets to represent linguistically described properties (i.e. labels) of objects in a database.

The interval type-2 linguistic summary of a database is a semi-natural language sentence

\[
Q \ P \text{ are/have} \ S \ [\underline{L}, \overline{T}] \quad (13)
\]

where the symbols \( Q, P, \) and \( S \) are interpreted as in (7), but \( S \) is modelled by an IT2FS, and \( T = [\underline{L}, \overline{T}] \subseteq [0, 1] \) is an interval-valued quality measure of the summary:

\[
T = \left[ \inf_{r \in [\underline{L}, \overline{T}]} \mu_Q(r), \sup_{r \in [\underline{L}, \overline{T}]} \mu_Q(r) \right] \quad (14)
\]

where

\[
[\underline{L}, \overline{T}] = \sum_{i=1}^{m} \left[ \mu_S(d_i), \overline{\mu}_S(d_i) \right] \quad (15)
\]

for an absolute quantifier (for a relative \( Q \), substitute \( r := \frac{r}{m} \)). If a summarizer \( S \) is to be composed from a few labels \( S_1, \ldots, S_n \), then

\[
\mu_S(d_i) = \min_{j=1,2,\ldots,n} \left\{ \mu_{S_j}(V_j(y_i)) \cdot \mu_{w_q}(V_q(y_i)) \right\} \quad (16)
\]

and \( \overline{\mu}_S(d_i) \) — analogously. It is also possible to process summaries with a query \( w_q \) represented by an ordinary fuzzy set:

\[
\mu_S(d_i) = \min_{j=1,2,\ldots,n} \left\{ \mu_{S_j}(V_j(y_i)) \cdot \mu_{w_q}(V_q(y_i)) \right\} \quad (17)
\]

and \( \overline{\mu}_S(d_i) \) — analogously. Now, the \( T \) index is

\[
T = \mu_Q \left( \frac{\sum_{i=1}^{m} \mu_S(d_i), \sum_{i=1}^{m} \overline{\mu}_S(d_i)}{\sum_{i=1}^{m} \mu_{w_q}(V_q(y_i))} \right) \quad (18)
\]

### 3.3 Quality measures for linguistic summaries

The indices presented in this section were first defined by Traczky [16] to determine the quality of knowledge mined from databases due to lengths of sentences expressing some facts, or due to shapes of fuzzy sets which represent chosen information. These indices were originally reformulated by Kacprzyk, Yager, and Zadrożny [5] and applied to linguistic summaries quality determining. Therefore, imprecision (denoted as \( T_2 \)), covering (\( T_3 \)), appropriateness (\( T_4 \)), and length (\( T_5 \)) of linguistic summaries are determined, since Yager’s "classic" \( T \) is denoted as \( T_1 \). The interval forms of these indices are given in [13]; here, they are interpreted in terms of IT2FSs.

#### Degree of truth

The interval-valued degree of truth is given by (14). In the case when \( \mu_S = \overline{\mu}_S \), its form is reduced to (8).

#### Degree of imprecision

The degree of imprecision, \( T_2 \), is a very intuitive criterion which describes how imprecise the summarizer used in a summary is. It is first required to define the degree of fuzziness of an IT2FS \( S_i \) in \( X \):

\[
in(S_i) = [\underline{in}(S_i), \overline{in}(S_i)] \quad (19)
\]
where
\[
\text{in}(S_i) = \frac{|\{x \in X_i : \text{I}_S(x) > 0\}|}{|X_i|} \tag{20}
\]
and \(\overline{\text{in}}(S_i)\) – analogously. The degree of imprecision is given as the interval \([\overline{L}_2, \overline{T}_2]\), where
\[
\overline{L}_2 = 1 - \left(\prod_{j=1}^{n} \overline{\text{in}}(S_j)\right)^{1/n} \tag{21}
\]
and \(\overline{T}_2\) – analogously. The semantics of this index is: the flatter \(\mu_{S_j}\), the closer to unity \(\text{in}(S_j)\) is, hence, the less precise the feature \(S_j\).

**Degree of covering** The degree of covering is based on the \(t_i\) function:
\[
t_i = \begin{cases} 1, & \text{if } \mu_{S_i}(d_i) > 0 \land \mu_{w}(V_g(y_i)) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{22}
\]
and \(\overline{t}_i\) – analogously, and on the \(h_i\) function,
\[
h_i = \begin{cases} 1, & \text{if } \mu_{w}(V_g(y_i)) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{23}
\]
Thus, \(T_3 = [\overline{L}_3, \overline{T}_3]\) is in the form of
\[
[\overline{L}_3, \overline{T}_3] = \left[\frac{\sum_{i=1}^{m} \overline{L}_i \cdot \sum_{i=1}^{m} \overline{T}_i}{\sum_{i=1}^{m} h_i}\right] \tag{24}
\]
The degree of covering determines how many objects in the database corresponding to the query \(w_g\) are covered by the summary.

**Degree of appropriateness** If a summarizer \(S\) is represented by the family of fuzzy sets \(\{S_1, ..., S_n\}\), it may be divided into \(n\) partial summaries based on attributes \(S_1, ..., S_n\), respectively. The degree of appropriateness is based on the \(r\) index computed for \(S_j\) as
\[
[S_j, \overline{T}_j] = \left[\frac{\sum_{i=1}^{m} g_i, \sum_{i=1}^{m} \overline{g}_i}{m}\right] \tag{25}
\]
where
\[
g_i = \begin{cases} 1, & \text{if } \mu_{S_j}(V_j(y_i)) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{26}
\]
and \(\overline{g}_i\) – analogously. The degree of appropriateness is equal \([L_j, \overline{T}_j]\), where
\[
L_j = \left[\prod_{j=1}^{n} L_j - \overline{T}_3\right] \tag{27}
\]
and \(\overline{T}_4\) – analogously. \(T_j\) is said to be the most relevant degree of validity of summaries.

**Length of summary** The index of quality called a *length of a summary*, denoted as \(T_5\), is defined as
\[
T_5 = 2 \cdot (0.5)^{\text{card}(S)} \tag{28}
\]
where \(\text{card}(S)\) is the number of features represented by single fuzzy sets. Thus, \(T_5\) indicates the following: the longer the summary is, the smaller its correctness\(^2\). All the presented indices can be used to determine a reliable quality measure for the summary:
\[
T(T_1, ..., T_5; w_1, ..., w_5) = \sum_{i=1}^{5} w_i \cdot T_i \tag{29}
\]
where: \(w_1 + w_2 + w_3 + w_4 + w_5 = 1\).

4 The algorithm

The schema of the system which generates textual messages on a given database, is depicted in Fig. 1.

![Figure 1: The news generator](image)

As it is seen, the process is not performed automatically; similarly to systems that support medical diagnosis, each artificially made decision must be corrected/verified/rejected by a human expert. Input data from a database, from user’s and expert’s entries, are processed by the summaries generator via the algorithms described in this section.

The symbolical form of the database is given by (6). Let \(\{Q_1, ..., Q_k\}\) – a set of \(k\) fuzzy quantifiers be given. Let a set of \(z\) fuzzy sets \(\{S_1, ..., S_z\}\) be given in the domains of attributes \(V_1, ..., V_z\), respectively. The number of possible summaries generated from \(z\)

\(^2T_5\) has no interval form; it is always a crisp number.
summarizers is:
\[
k(\tilde{i}) \left[ \tilde{i} \cdots \tilde{i} + \cdots + \tilde{i} \right] + \\
+k(\tilde{i}) \left[ (i-1) \cdots + (i-1) \right] + \cdots + \\
+k(\tilde{i}) \left[ \left( \frac{1}{i} \right) \right] = \\
= k(\tilde{i}) (2^{i-1}) + \cdots + k(\tilde{i}) (2^{i-1}) = \\
= k \sum_{i=0}^{\infty} (\tilde{i})(2^{i-1} - 1)
\]
(30)

We treat Yager’s form (Y) of a summary as a special case of George and Srikanth’s form (G). In consequence, Kacprzyk and Zadrozny’s (K) form of a summary (with a \( w_g \) query) is a further generalization of (G), or, in other words, (G) is a special case of (K) in which \( w_g = \emptyset \). Hence, the form of the first element, \( (\tilde{i}) \left[ \tilde{i} + \cdots + \tilde{i} \right] = \emptyset (2^{i-1} - 1) \), results from the fact that we choose 0 of z summarizers to build \( w_g \) and \( i = 1, \ldots, z \) summarizers to build \( S \). The next one, \( (\tilde{i}) (2^{i-1} - 1) \), is related to the choice of exactly one summarizer for \( w_g \) and \( i = 1, \ldots, z - 1 \) summarizers for \( S \), etc.

Formula (30) is valid only if the AND connective is used to build composite summarizers or queries, e.g. “high salary AND young”; see (5) and (9). Nevertheless, it is worth mentioning that summarizers can be also joined with other connectives, like OR, etc.

// generating (Y) summaries
1. for each single summarizer \( S \in \{S_1, \ldots, S_z\} \)
   1.1. for each quantifier \( Q_h, h = 1, \ldots, k \)
       if \( Q_h \) is absolute
           compute \( T_h = \mu_{Q_h} \left( \sum_{d \in D} \mu_{s_k}(d) \right) \)
       else i.e. if \( Q_h \) is relative
           compute \( T_h = \mu_{Q_h} \left( \sum_{d \in D} \mu_{s_k}(d) \right) \)
   1.2. compute \( T_{h_{\text{max}}} = \max_{h=1, \ldots, k} T_h \), remember \( h_{\text{max}} \)
   1.3. generate summary in the form of \( Q_{h_{\text{max}}} P \) is/have \( S \) \([T_{h_{\text{max}}}]\)

// generating (G) summaries
2. for each non-singleton and non-empty \( S \subseteq \{S_1, \ldots, S_z\} \)
   2.1. determine \( \bar{\mu}(d_j) = \min_{S_j \in S} \mu_{S_j}(d_j) \)
   2.2. for each quantifier \( Q_h, h = 1, \ldots, k \)
       compute \( T_{h_{\text{max}}} \) analogously to 1.1.
   2.3. compute \( T_{h_{\text{max}}} = \max_{h=1, \ldots, k} T_{h_{\text{max}}} \)
       remember \( h_{\text{max}} \)
   2.4. compute
       \[
       T_2 = 1 - \left( \prod_{S_j \in S} \frac{\text{card}(\{x \in X_j \mid \mu_{S_j}(x) > 0\})^{1/\text{card}(S)}}{\text{card}(X_j)} \right)
       \]
       // \( T_3 = 0 \) since \( w_g = \emptyset \)
   2.5. compute \( T_4 = \left[ \prod_{S_j \in S} \sum_{d \in D} \mu_{S_j}(d) \right] \)

where \( g \) is given by (26)
2.6. compute \( T_5 = 2 \cdot \left( \frac{1}{\text{card}(S)} \right) \)
2.7. compute
   \[
   T = w_1 \cdot T_{h_{\text{max}}} + w_2 \cdot T_2 + w_4 \cdot T_3 + w_5 \cdot T_5
   \]
2.8. generate summary in the form of \( Q_{h_{\text{max}}} P \) are/have \( S \) \([T]\)

// generating (K) summaries
3. for each non-empty query \( S_w \subset \{S_1, \ldots, S_z\} \) and for each non-empty summarizer \( S \subseteq \{S_1, \ldots, S_z\} \backslash S_w \)
   3.1. determine \( \bar{\mu}_j(d_j) = \min_{S_j \in S} \mu_{S_j}(d_j) \)
   3.2. determine \( D \supseteq D_w = \{d \in D : \mu_{S_w}(d) > 0\} \)
   3.3. for each \( d \in D_w \)
       determine \( \bar{\mu}_j(d_j) = \min_{S_j \in S} \mu_{S_j}(d) \)
   3.4. for each \( h = 1, \ldots, k \) compute
       \[
       T_{1, h} = \mu_{Q_h} \left( \frac{\sum_{d \in D_w} \mu_{S_w}(d)}{\text{card}(D)} \right)
       \]
   3.5. choose \( T_{1, h_{\text{max}}} \) analogously to step 1.2.
   3.6. compute \( T_2 \) analogously to 2.4.
   3.7. compute \( T_3 = \sum_{d \in D_w} \mu_{S_w}(d) \)
       where \( t \) and \( h \) are given by (22) and (23), resp.
   3.8. compute \( T_4 = \prod_{S_j \in S} \sum_{d \in D_w} \mu_{S_j}(d) \)
   3.9. compute \( T_5 \) analogously to 2.6.
   3.10. compute \( T = T_{h_{\text{max}}} + \sum_{i=2}^{4} w_i T_i \)
   3.11. generate summary in the form of \( Q_{h_{\text{max}}} P \) having \( S_w \) are/have \( S \) \([T]\)

Ad. 2. In generating (G) summaries which are based on at most \( z \) summarizers, the algorithm which enables determining all the non-singleton and non-empty subsets of \( \{S_1, \ldots, S_z\} \) is required; the number of such subsets is exactly \( 2^z - 1 \). In the implementation, the problem is resolved via generating binary forms of all natural numbers between 0 and \( 2^z - 1 \); the forms are taken as characteristic vectors of the sought subsets.

Ad. 2.4. The \( X_i \) set is the domain of the \( S_i \) fuzzy set.

Ad. 2.6. \( w_1 + w_2 + w_4 + w_5 = 1 \).

Ad. 3.10. \( w_1 + w_2 + w_3 + w_4 + w_5 = 1 \).

Other variants of the algorithm, e.g. commenting on a few attributes with respect to all their values are described in [14] from the point of view of ordinary fuzzy sets. Extensions and more general approaches (for type-2 fuzzy sets with different secondary MFs) are currently being developed.
5 Implementation

The prototype system is implemented on .NET platform in the C# language. A sample database consists of 5000 records in the form of \(<\text{ID}, \text{practice}, \text{age}, \text{salary}>\) and is implemented with MS SQL Server. Three interval type-2 fuzzy summarizers are applied to interpret numerical values in columns, respectively: \(S_1 = \text{"long"}, S_2 = \text{"middle-aged"}, S_3 = \text{"average"}\) with trapezoidal lower and upper membership functions \((z = 3)\). Three type-1 fuzzy quantifiers \((k = 3)\) are determined: two in the \([0, 1]\) domain (relative quantifiers): \(Q_1 = \text{"few"}, \mu_{Q_1}(x) = x, Q_2 = \text{"about half"}, \mu_{Q_2}(x) = -|2x - 1| + 1,\) and one absolute quantifier \(Q_3 = \text{"much more than 2000"}\):

\[
\mu_Q(x) = \begin{cases} 
0, & \text{if } x \leq 2000 \\
\frac{x - 2000}{500}, & \text{if } 2000 \leq x \leq 2500 \\
1, & \text{otherwise}
\end{cases}
\]

All the quantifiers are represented by ordinary fuzzy sets. The weights for the quality measures are set as \(w_1 = w_2 = w_4 = w_5 = \frac{1}{4}\) for \((G)\) summaries and \(w_1 = w_2 = w_3 = w_4 = w_5 = \frac{1}{5}\) for \((K)\) summaries.

The number of expected summaries is, according to (30), \(3 \binom{3}{1}(2^3 - 1) + 3 \binom{3}{2}(2^2 - 1) + 3 \binom{3}{3}(2^1 - 1) = 57\). The final message consists of 16 sentences, i.e. \(3(Y) + 4(G) + 9(K)\) summaries.

The sample message generated via the algorithm described in Section 4 is presented below:

About half of employees are middle-aged \([0.42, 0.54]\). Much more than 2000 employees have long practice and are middle-aged \([0.32, 0.38]\). (...) Few of employees having long practice have average salary \([0.88, 1.00]\). Few of employees having average salary have long practice \([0.40, 0.57] (...)\)

The obtained message is more informative than the similar one produced with type-1-fuzzy-based (Yager’s) methods. At first, although it remains invisible for an end-user, the summarizers applied, e.g. middle-aged or average salary, are suggested by more than one expert, hence they are more objective. Moreover, the qualities of given summaries are expressed with interval numbers, which is supposed to give better and more human consistent information about goodness levels of the messages generated – the intervals provide information about the range in which goodness of summaries can be considered (e.g. optimistic and pessimistic variants let the user choose the upper or the lower bound of a given interval, respectively, as a quality).

6 Conclusions and further work directions

A new extension of linguistic summaries of databases in sense of Yager has been introduced – the extension is based on Interval Type-2 Fuzzy Sets instead of ordinary fuzzy sets. Moreover, the presented algorithms may establish the point of departure to further generalizations of methods of linguistic summarizing (e.g. via triangular or trapezoidal T2FSs which are more flexible than Interval T2FSs); some attempts have already been made in [11] and are currently being developed.

The presented algorithms mechanize reporting of large datasets and provide linguistically formulated and user-friendly results. The prototype implementation and sample outputs have been presented.

References


