Computing with Words for Direct Marketing Support System

Pakizar Shamoi

Department of Computer Science,
Kazakh-British Technical University
pakita883@gmail.com

Atsushi Inoue

Department of Computer Science,
Eastern Washington University
inoueatsushij@gmail.com

Abstract

This paper highlights the simplicity and effectiveness of Computing with Words (CW) in the implementation of target selection. Direct marketing can be considered as one of the main areas of application for this methodology. In particular, fuzzy classification is applied in it with the purpose of choosing the best potential customers for a new product or service from a client database. One of the advantages of the proposed method is that it is consistent with relational databases. Our methodology makes it possible to form queries in natural language, such as "*print the list of not very old married clients with more-or-less high income*", which is impossible using a standard query mechanism.

Introduction

There is one fundamental advantage of humankind that needs to be inculcated into the various information systems. It is the remarkable human ability to perform a wide range of mental tasks without any measurements and any computations (Zadeh, 2002; Herrera, et al., 2009; Martinez, et al., 2010). That is possible due to the brain's crucial ability to manipulate perceptions of size, distance, weight, speed, etc. (Zadeh, 2002). The main difference between measurements and perceptions is that the former are crisp whereas the latter are vague (fuzzy) - the transition from membership to non-membership is gradual rather than sudden.

The main purpose of using natural (linguistic) queries instead of numbers is that it is much closer to the way that humans express and use their knowledge. Perception-based rational decisions in an environment of imprecision are becoming highly actual (Zadeh, 2002). An important use of the Computing with Words (CW) methodology, which is in the heart of fuzzy logic, is its application to decision making (Zadeh, 1965; Zadeh, 1975; Zadeh, 1996; Ying, 2002; Herrera, et al., 2009). In fact, CW can simplify the decision processes when the experts can only provide qualitative, but not quantitative information about the evaluated alternatives (Herrera, et al., 2009).

This paper is organized in six sections. First one is this introduction. Next we emphasize the critical importance of target selection in direct marketing. Furthermore, we examine in details the how fuzzy approach was applied to make the process of target selection more efficient. Particularly, it discusses the concepts of linguistic variables and hedges, fuzzification, explains the need for imprecision. The next section considers in detail the fuzzy querying model implemented using Computing with Words methodology that is solely based on fuzzy mathematics. Then we provide illustrative examples of different types of queries and their result sets obtained from the application developed. Finally, the last section provides the concluding remarks of this study.

The Role of Target Selection in Direct Marketing

Direct marketing (DM) is a form of advertising that enables companies to communicate directly to the customer, with various advertising techniques including email, mobile messaging, promotional letters, etc. The crucial idea there is to be able to deliver the marketing message to the clients that are likely to be interested in the product, service, or offer (Mederia and Sousa, 2002). So, DM companies or organizations try to set and maintain a direct relationship with their clients in order to target them individually for specific product or service.

An important data mining problem from the world of DM is target selection (Mederia and Sousa, 2002). The main task in target selection is the determination of potential customers for a product from a client database. Target selection algorithms identify the profiles of customers who are likely to respond to the offer for a particular product or service, given different types of information, like profession, age, purchase history, etc. In addition to the numerical performance, model transparency is also important for evaluation by experts, obtaining confidence in the model derived, and selecting an appropriate marketing channel (Mederia and Sousa, 2002). Fuzzy models for target selection are interesting from this angle of view, since they can be used to obtain numerically consistent models, while providing a linguistic description as well.

As mentioned above, in DM the selection of the target audience is a very important stage. Different DM techniques benefit from accurate target selection. Take, for example, a direct mail, used in the promotion of goods and services to organizations and individuals through electronic mail. Some DM methods using particular media, especially email have been criticized for poor target selection strategy. This poses a problem for marketers and consumers alike. On the one hand, advertisers do not wish to waste money on communicating with consumers not interested in their products. Also, they don’t want to lose potential customers. On the other hand, people usually try to avoid spam. However, they want to be aware of the new products/services that might be interesting for them.

As previously mentioned, in order to maximize its benefits direct mail requires careful selection of recipients. So, if the selection of recipients is too liberal, it will increase unnecessary spending on DM, if it is too strict – we’ll lose some potential customers. Virtually all companies that work with a database of 100 or more customers use email-mailing in their business (Ribeiro and Moreira, 2003). But again, this is a very delicate instrument, because the line between a useful message and spam is very thin. Therefore, companies providing mailing services, must constantly engage in outreach efforts, so that due to their own ignorance, they do not lose customers and reputation, sending spam and making other common mistakes.

Fuzzy Approach

Need for imprecision

Nowadays, most of the data processed in information systems has a precise nature. However, a query to the database, formed by a person, often tends to have some degree of fuzziness. For example**,** the result of a query in a search engine is a lot of references to documents that are ordered by the degree of relevance to the request. Another simple example of natural query used in everyday life: "Find a listing for housing that is *not very expensive* and is *close* to downtown". Statements like *"not very expensive," "close"* are vague, imprecise, although rent price is completely determined, and the distance from the center of the apartment - up to a kilometer. The cause of all these problems is that in real life, we operate and argue using imprecise categories (Zadeh, 1965; Zadeh, 1975).

For example, a company launches an advertising campaign among their clients about new services through direct mail. The Marketing Service has determined that the new service will be most interesting for *middle-aged married men*, with *more-or-less high income*. A crisp query, for example, might ask for all married males aged 40 to 55 with an income of more than 150,000 tenge. But with such a request we may weed out a lot of potential clients: a married man aged 39, with an income of 250,000 tenge does not fall into the query result, although he is a potential customer of the new service.

Linguistic variables

One of the main hindrances of modern computing is that a concept cannot be well understood until it is expressed quantitatively. This is where linguistic variables come in. The main motivation to prefer linguistic variables rather than numbers is that a linguistic description is usually less specific than a numerical one (Zadeh, 1975).

According to Zadeh , “By a linguistic variable we mean a variable whose values are not numbers but words or sentences in a natural or artificial language” (Zadeh, 1975). So, for example, *Income* is a linguistic variable if its values are linguistic (*not very low, average, more-or-less high,* …) rather than numerical (100 000 tg., 150 600 tg.…). Following that logic, the label *high* is considered as a linguistic value of the variable *Income*, it plays the same role as some certain numerical values. However, it is less precise and conveys less information.

To clarify, in the example provided, *Income* is a linguistic variable, while *low, average, high* are linguistic values, represented in the form of fuzzy sets. The set of all linguistic values of a linguistic variable is called term set.

Although a linguistic value is less precise than a number it is closer to human cognitive processes, and that can be exploited successfully in solving problems involving uncertain or ill-defined phenomena. So, in situations where information is not precise (which are very common in our real life), linguistic variables can be a powerful tool that takes the human knowledge as model (Herrera and Herrera-Viedma, 2000).

Besides their primary meaning, linguistic values may involve connectives such as *and, or, not* and hedges such as *very, quite extremely, more or less, completely, fairly*, etc. about which we will talk extensively later.

Fuzzification

It is highly important for any target selection model to select the clients’ features that will play the role of explanatory variables in the model (Mederia and Sousa, 2002). They serve to reflect the essential characteristics of the clients that are important for the product or service and they vary from organization to organization. Therefore, just for the sake of simplicity in this model we present just some of the possible criteria – gender, age, status, income.

So, let’s suppose we have a table “Clients”, consisting of 7 rows: id (primary key), name, gender (‘Male’,‘Female’), age, status (‘Married’, ‘Not_married’), email, income.

Table 1. Structure of the sample table for the system

Field	Type	Fuzzy	Comments
id	**int**		**auto increment**
name	**varchar**		
gender	**enum**		**('Male', 'Female')**
age	**int**	√	
status	**enum**		**('Married', 'Not_married')**
email	**varchar**		
income	**int**	√	

By the way, in practice, a certain threshold of membership value is given in excess of which records are included in the result of a fuzzy query. Usually it is a number between 0 and 1 and can be represented to the user in the form of percentage. So, an expert can manoeuvre with it to make the query more or less strict. One of the situations in which a threshold can be very efficient is when expert receives a long list of clients as a response to a query. Then, he can decide to be stricter and make the threshold higher in order to be more confident in the buying power of the clients.

Usually, in real-world decision making processes there are experts - decision makers who choose the appropriate initial parameters to define the fuzzy variables (Martinez, et al., 2010; Herrera and Herrera-Viedma, 2000). So, because of different cultural reasons or different points of view and knowledge about the problem it seems reasonable to give possibility to decision makers to provide their preferences about the problem on their own. That is why a, b, and c parameters for each of the fuzzy variables should be input to the system by the expert. Again, this is done, since it seems difficult to accept that all experts should agree to the same membership functions associated with primary linguistic terms.

Many decision problems need to be solved under uncertain environments with blurred and imprecise information. The use of linguistic information in decision making involves processes of CW (discussed a bit later).

Fuzzy sets and logic play a major role in this project. Fuzzy mathematics allows us to use the imprecision in a positive way. It is very efficient in complex problems that can't be handled using standard mathematics, like processing human elements – natural language, perception, emotion, etc. The term "fuzzy" can be defined as "not clear, blurred, or vague." For example, the word "tall" is fuzzy, since it is subjective term. For some people, man with the height 190 cm (6.2 feet) is tall, whereas for others 170 cm (5.7 feet) is enough to call the person "tall". As Zadeh said, "Fuzzy logic is determined as a set of mathematical principles for knowledge representation based on degrees of membership rather than on the crisp membership of classical binary logic" (Zadeh, 1996). According to traditional boolean logic, people can be either tall or not tall. However, in fuzzy logic in the case of the fuzzy term "tall," the value 170 can be partially true and partially false. Fuzzy logic deals with degree of membership with a value in the interval [0, 1]. In this paper fuzzy sets are used to describe the clients' age and income in linguistic terms which are fuzzy variables.

A computationally efficient way to represent a fuzzy number is to use the approach based on parameters of its membership function. Linear trapezoidal or triangular membership functions are good enough to catch the ambiguity of the linguistic assessments (Herrera and Herrera-Viedma, 2000). It is not necessary to obtain more accurate values. The proposed parametric representation is achieved by the 3-tuple (a; b; c) for each fuzzy variable, it is enough for 3 fuzzy sets, since we applied a fuzzy partition.

Now let's try to formalize the fuzzy concept of the client's age. This will be the name of the respective linguistic variable. We define it for the domain X = [0, 90], so, the universal set U = {0 ,1, 2, …..,89, 90}. The term set consists of 3 fuzzy sets – {"Young", "Middle-aged", "Old"}.

The last thing left to do - to build certain membership functions belonging to each linguistic term – fuzzy set We define the membership functions for the young, middle-aged, and old fuzzy sets with the following parameters [a,b,c] = [18,35,65]. In general form they look like:

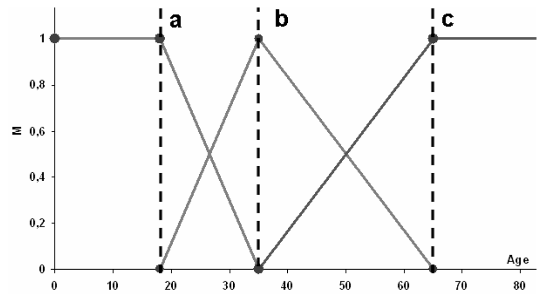

Figure 1. Fuzzy sets for young, middle-aged, and old

$$\text{"Young"} = \begin{cases} \mathbf{1}, & x \le a \\ \mathbf{1 - (x - a) / (b - a)}, & a \le x \le b \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

$$\text{"Middle-aged"} = \begin{cases} \mathbf{1 - (b - x) / (b - a)}, & a \le x \le b \\ \mathbf{1 - (x - b) / (c - b)}, & b \le x \le c \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

$$\text{"Old"} = \begin{cases} \mathbf{0}, & x \le b \\ \mathbf{1 - (c - x) / (c - b)}, & b \le x \le c \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

Figure 2. Membership functions for young, middle-aged, and old

Now we can, for example, calculate the degree of membership of a 30-year-old client in each of the fuzzy sets:

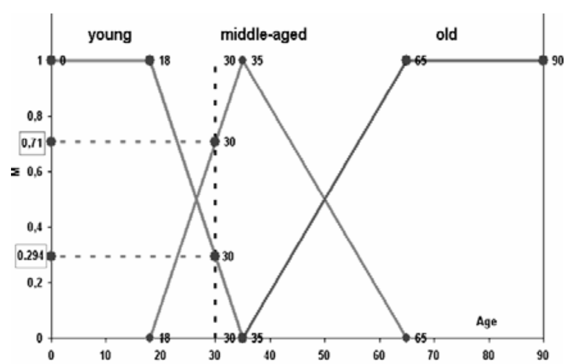

Figure 3. Membership of a 30-year-old client in young, middle-aged, and old. (μ [Young] (30) = 0,294, μ [Middle-aged] (30) = 0,71, μ [Old] (30) = 0).

Another fuzzy variable in the system is client's income. We define it for the domain X = [0, 1000 000], so, the universal set U = {0 ,1,…, 250 000, …,1 000 000}. The term set consists of 3 fuzzy sets – {"Low", "Average", "High"}. The membership functions for income variable term set are totally similar to the ones discussed above. The parameters are the following [a,b,c] = [40 000, 100 000, 200 000]. Income fuzzy variable, so as Age, is partitioned by three fuzzy sets associated with linguistic labels. Each fuzzy set corresponds to perception agents – low, average, or high salary. As it can be seen from the graph, there are no sharp boundaries between low, average, and high.

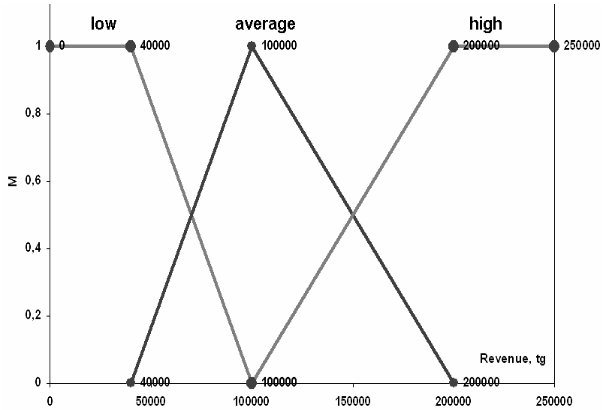

Figure 4. Fuzzy Sets for low, average, and high income. A fuzzy partition.

It is highly important to remember that decision making is an inherent human ability which is not necessarily based on explicit assumptions or precise measurements. For example, typical decision making problem is to choose the best car to buy. Therefore, fuzzy sets theory can be applied to system to model the uncertainty of decision processes.

Linguistic Hedges

Knowledge representation through linguistic variables characterized by means of linguistic modifiers – hedges makes the query more natural, so their main advantage is the ability to be expresses in natural language. Hedges can change the statement in various ways – intensify, weaken, complement. Their meaning implicitly involves fuzziness, so their primary job is to make things fuzzier or less fuzzy.

Let's consider the most common ways of generating new fuzzy sets based on the initial fuzzy set using various hedges. This is useful for constructing various semantic structures -composite words - from atomic words (i.e. *young*) that reinforce or weaken the statements (Zadeh, 1996) such as *very high salary, more-or-less old*, etc.

Again, the main motivation is to strengthen or weaken the statement (Zadeh, 2002). For reinforcing there is the modifier *very*, to weaken - *more-or-less or almost*, *approximately*. Fuzzy sets for them are described by certain membership functions. Hedges can be treated as operators which modify the meaning in a context-independent way.

For example, let's suppose that the meaning of *X* (*middle-aged*) is defined by some membership function. If we want to strengthen the statement, we use *very* intensifier (Zadeh, 2002). Then the meaning of *very X* (i.e. very middle-aged) could be obtained by squaring this function:

$$\mu F_{VERY}(X) = (\mu F(X))^2$$

Figure 5 demonstrates that *very* hedge steepens the curve.

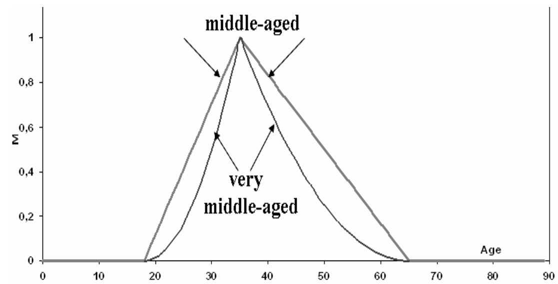

Figure 5. Visualizing the hedge *very*

Furthermore, the modifier that can weaken the statement - *more-or-less X* (i.e. more-or-less middle-aged) would be given as a square root of the initial function (Zadeh, 2002):

$$\mu F_{MORE-OR-LESS}(X) = \sqrt{\mu F(X)}$$

Figure 6 illustrates that *more-or-less* hedge makes the curve less steep.

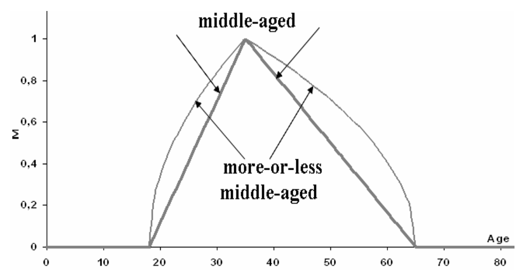

Figure 6. Visualizing the hedge *more-or-less*

Finally, *not X* (i.e. not young) which is a complement fuzzy set, can be expressed by subtracting the membership function of *X (middle-aged)* from 1:

$$\mu F_{NOT}(X) = 1 - \mu F(X)$$

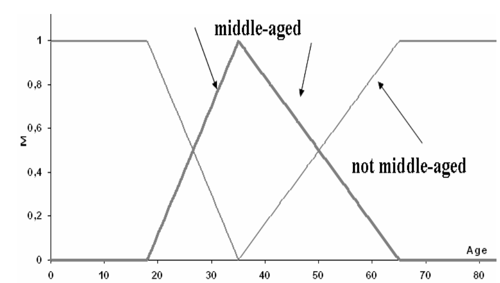

Figure 7. Visualizing the modifier *not*

Let's enjoy calculating the membership of 30-year-old client to each of the fuzzy sets: *middle-aged, not middle-aged, very middle-aged,* and *more-or-less middle-aged.*

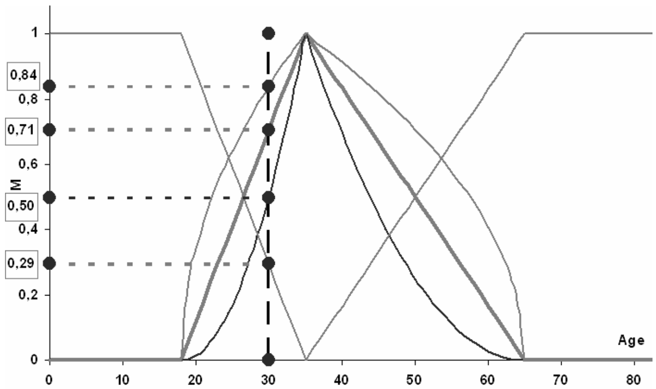

Figure 8. μ [middle-Aged] (30) = 0,71; μ [not middle-aged] (30) = 0,29; μ [very middle-aged] (30) = 0,5; μ [more-or-less middle-aged] (30) = 0,84.

As we have seen, hedges intrinsically convey the imprecision in themselves. The main flexibility they provide is that they can make a fuzzy natural query even more natural.

Computing with Words

Computing with words (CW), originally developed by Zadeh, provides a much more expressive language for knowledge representation. In it, words are used in place of numbers for computing and reasoning, with the fuzzy set playing the role of a fuzzy constraint on a variable (Zadeh, 2002). CW is a necessity when the available information is too imprecise to justify the use of numbers, and when there is a tolerance for imprecision that can be exploited to achieve tractability, robustness, low solution cost, and better rapport with reality (Zadeh, 1996).

A basic premise in CW is that the meaning of a proposition, *p*, may be expressed as a generalized constraint in which the constrained variable and the constraining relation are, in general, implicit in *p* (Zadeh, 1996). In the system proposed here the CW methodology is slightly adapted and modified in order to be able to process natural queries, not propositions.

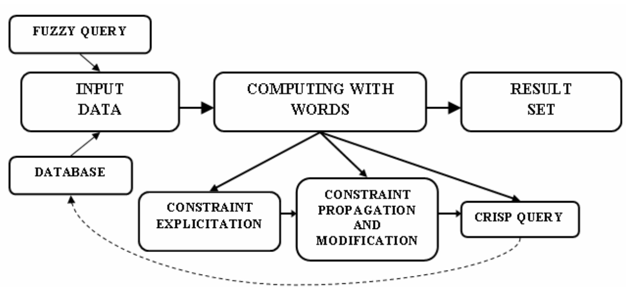

Figure 9. CW approach to target selection

CW methodology is changed a little bit to correspond to the proposed system. In particular, the initial data set (IDS) is a database with clients' information. From the IDS we desire to find the subset of clients from the database in response to a query expressed in a natural language. That is our result - terminal data set (TDS). So, our goal is to derive TDS from IDS. In our model, we process the natural query step by step, by constraining the values of variables, this process will be considered in details later.

Our aim is to make explicit the implicit fuzzy constraints which are resident in a query. So, how can we make explicit the fuzzy constraints that are given in natural language and so as are implicit?

In linguistic features of variables, words play the role of the values of variables and serve as fuzzy constraints at the same time (Zadeh, 1996). For example, the fuzzy set *young* plays the role of a fuzzy constraint on the age of clients. *Young* takes the values with respect to certain membership function. In a very general case query consists from a set of criteria and set of constraints put on those criteria. So far, we have primary terms for income and age - *high, average, low, young, middle-aged, old*; hedges - *not, very, more-or-less*; connectives - *and, but, or*.

In outline, a query *q* in a natural language can be considered as a network of fuzzy constraints. After processing procedure we get a number of overall fuzzy constraints, which can be represented in the form *X* is *R*, *Y* is S…, where *X* is a constrained criterion variable (i.e. age) which is not explicit in *q,* and R is a constraint on that criterion. So the explicitation process can be defined as:

$$q \rightarrow X \text{ is } R,\ Y \text{ is } S\ldots, \text{etc.}$$

As a simple illustration, let's consider the simple query: *not very young males with more-or-less high income.* As we can observe, some of the variables in a query are crisp, while some have fuzzy constraints. Let's assume that the user chose the threshold value μ_{Total} as a sufficient level of precision. So, we obtain:

YOUNG[Age; not, very; μ_{Total}**]** $\cap$ **HIGH[Income; more-or-less;** μ_{Total}**]** $\cap$ **MALE[Gender; ;** $\mu_{\text{Total}} = 1$**]**

Notice that *not very young* and *very not young* are different things. Therefore, the order is important. Another main point is that μ_{Total} is a membership value reflecting the degree of membership to *not very young* and *more-or-less high,* not to young and high. We need to pay special attention to it. To obtain the answer we need the membership value that corresponds to *young* and *high*, of course. That is why, the process is reversed: before we presented the formulas to shift to *very young* from *young*. Now, instead, we want to define *young* using *very young*. So, if we squared the threshold for *young* to get the threshold for *very young*, now we apply the inverse operation – square root. Furthermore, we get:

YOUNG[Age; very; 1- μ_{Total}**]** $\cap$ **HIGH[Income; ;** μ_{Total}^{2}**]** $\cap$ **MALE[Gender; ;** $\mu_{\text{Total}} = 1$**] = YOUNG[Age; ;** $\sqrt{1-\mu_{\text{Total}}}$**]** $\cap$ **HIGH[Income; ;** μ_{Total}^{2}**]** $\cap$ **MALE[Gender; ;** $\mu_{\text{Total}} = 1$**]**

Let's consider the translation rules that can be applied singly or in combination (Zadeh, 1996). These translation rules are:

a) Modification rules. Example: '*very old'*;

b) Composition rules. Example: *'young and more-or-less middle*-aged' ;

Constraint Modification Rules. X is *mA* → *X* is *f* (*A*) where *m* is a modifier – hedge or negation (very, not, more-or-less), and *f* (*A*) defines the way *m* modifies *A.*

It should be stressed that the rule represented is a convention and shouldn't be considered as the exact reflection of how *very, not or more-or-less* function in a natural language (Zadeh, 1996). For example, negation *not* is the operation of complementation, while the intensifier *very* is a squaring operation:

$$\text{if } m = not \text{ then } f(A) = A'$$
$$\text{if } m = very \text{ then } f(A) = A^2$$

Constraint Propagation Rules. Constraint propagation plays crucial role in CW. It is great that all the stuff with numbers takes plays outside of a user's vision.

The rule governing fuzzy constraint propagation: If *A* and *B* are fuzzy relations, then disjunction – *or* (union) and conjunction – *and* (intersection) are defined, respectively, as max and min (Zadeh, 1996).

Users can express the intersection in 3 ways distinguished by the connective type – *and, but*, or no connective at all. As it was previously stated, for this operation, we take the minimum of two memberships to get the resultant membership value:

$$\mu A(x) \cap B(x) = \min[\mu A(x), \mu B(x)]$$

Union is represented solely by *or* connective. The resultant membership value is equal to the maximum of two values provided:

$$\mu A(x) \cup B(x) = \max[\mu A(x), \mu B(x)]$$

The threshold used in the system serves as the α-cut (Alpha cut), which is a crisp set that includes all the members of the given fuzzy subset f whose values are not less than α for 0< α ≤ 1:

$$f_\alpha = \{x : \mu_f(x) \geq \alpha\}$$

We also know how to connect α-cuts and set operations (let *A* and *B* be fuzzy sets):

$$(A \cup B)_\alpha = A_\alpha \cup B_\alpha, \; (A \cap B)_\alpha = A_\alpha \cap B_\alpha$$

So, using the formulas provided above, in order to find the result of a query with a certain threshold – α, containing *or* or *and* operations, we first find the α-cuts and then take the crisp or / and operation.

In dealing with real-world problems there is much to be gained by exploiting the tolerance for imprecision, uncertainty and partial truth. This is the primary motivation for the methodology of CW (Zadeh, 2002).

Application and Examples

More and more employees are depending on information from databases to fulfill everyday tasks. That is why nowadays it is becoming increasingly important to access information in a more human-oriented way – using natural language.

We presented a fuzzy querying model capable of handling various types of requests in a natural language form. The interface developed allows experts to express questions in natural language and to obtain answers in a readable style, while modifying neither the structure of the database nor the database management system (DBMS) query language.

The main advantage of our model is that the query to the system is done in a natural language. Besides, existing clients databases do not have to be modified and developers do not have to learn a new query language. Basically, we just have a fuzzy interface that is used as a top layer, on an existing relational database, so, no modifications on its DBMS were done.

The main problem in developing human-oriented query interfaces was how to allow users to query databases in a natural way. The motivation for that is that usually users don't wish to define the clear bounds of acceptance or rejection for a condition, that is, they want to be allowed some imprecision in the query (Ribeiro and Moreira, 2003).

The past and new conceptual structure of the model is schematized and illustrated below, in figure 10.

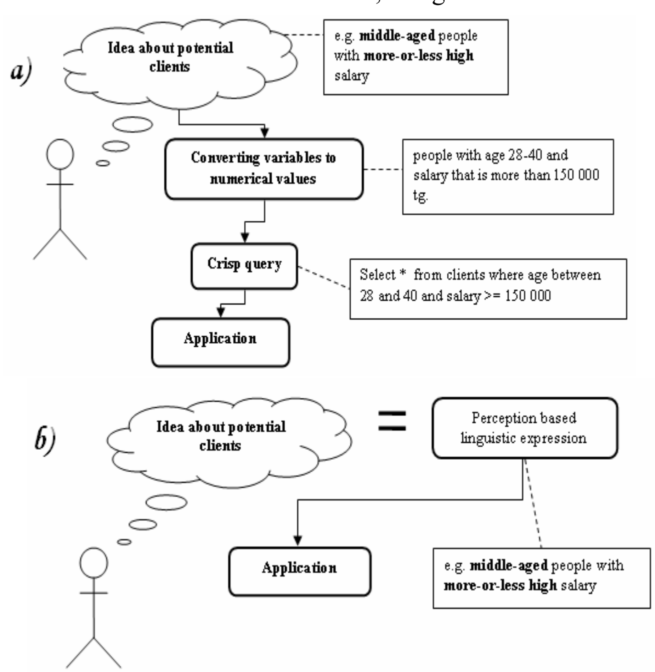

Figure 10. a) Traditional approach. b) CW approach

Let's look at examples of natural language queries given with the purpose of demonstrating the capabilities of this human-oriented interface.

The application interface is very friendly. If some user experiences problems forming a natural query, he can use another menu provided. In it the criteria are listed on the screen, and user can just pick and choose which ones he wants. Furthermore, the parameters he chooses appear and he needs to choose needed values ("*very young*", "*not old*", etc.) on the respective pull-down menu. Moreover, in order to adapt this model to other databases we won't need to change the logic, because it is context-independent. We will just need to change the list of fuzzy variables.

Example query 1. *not old* married males with *very high* income. [Threshold value: 0.5]

Here we have two crisp criteria - status is married, gender is male. Furthermore, there are two fuzzy criteria – age is *not old* and income is *very high.* So, we have:

OLD[Age; not; μ_{Total}=0.5] $\cap$ HIGH[Income; very; μ_{Total}=0.5] $\cap$ MALE[Gender; ; μ_{Total} = 1] $\cap$ MARRIED[Status; ; μ_{Total} = 1] = OLD[Age;; μ_{Total}=0.5] $\cap$ HIGH[Income; very; $\mu_{\text{Total}} \approx 0.7$] $\cap$ MALE[Gender; ; μ_{Total} = 1] $\cap$ MARRIED[Status;; μ_{Total} = 1]

Next our system finds the values of age and income that correspond to the thresholds obtained. For the age, the constraining relation will be "≤ 50", for the income – "$\geq$ 170 710 tg.".

Now, having a look at our sample table, we can find 2 clients, fully satisfying the query. The system gives us the same result:

id	name	gender	age	status	email	income
5	Ernar M.	Male	32	Married	era@gmail....	890 009
8	Karl L.	Male	50	Married	karl@hotm...	200 300

Example query 2. *middle-aged* but *not more-or-less old* clients. [Threshold value: 0.5]

Here we need to make the conjunction of two constraints on one fuzzy variable – age:

MIDDLE-AGED[Age; ; μ_{Total}=0.5] $\cap$ OLD[Age; not, more-or-less; μ_{Total}=0.5] = MIDDLE-AGED[Age; ; μ_{Total}=0.5] $\cap$ OLD[Age; ; μ_{Total}=0.25]

We obtain the following result (note, that if we queried just for middle-aged, then 50-yeared client Karl L. would be included to the result set):

id	name	gender	age	status	email	income
4	Iliyas T.	Male	28	Not_mar..	iliyas@gma...	305 000
5	Ernar M.	Male	32	Married	era@gmail....	890 009
6	Kamin...	Female	40	Married	kaminari@...	55 000
11	Madina..	Female	34	Not_mar…	madina_@...	30 000

Example query 3. *not very young* married clients with *average* or *more-or-less high* salary. [Threshold value: 0.7]

We obtain the following:

YOUNG[Age; not, very; μ_{Total}=0.7] $\cap$ MARRIED[STATUS; ; μ_{Total}=1] $\cap$ (AVERAGE[Income; ; μ_{Total}=0.7] $\cup$ HIGH[Income; more-or-less ; μ_{Total}=0.7]) = YOUNG[Age; ; $\mu_{\text{Total}} \approx$ 0.55] $\cap$ MARRIED[STATUS; ; μ_{Total}=1] $\cap$ (AVERAGE[Income; ; μ_{Total}=0.7] $\cup$ HIGH[Income; ; μ_{Total}=0.49])

The result set is the following:

id	name	gender	age	status	email	income
5	Ernar M.	Male	32	Married	era@gmail....	890 009
8	Karl L.	Male	50	Married	karl@hotm...	200 300
9	Amina L.	Female	74	Married	amina@ya...	120 000
13	Alfi A.	Male	67	Married	alfi@gmail...	88 000

Last thing to note, the hedges can be applied infinitely in any order! In order to demonstrate that in practice, consider the following example.

Example query 4. *very very very* old or *very very very* young**.** [Threshold value: 0.5]

OLD[Age; very, very, very ; μ_{Total}=0.5] $\cup$ YOUNG[Age; very, very, very; μ_{Total}=0.5] = OLD[Age; ; $\mu_{\text{Total}} \approx$ 0.92] $\cup$ YOUNG[Age; ; $\mu_{\text{Total}} \approx$ 0.92]

The targeted clients are:

id	name	gender	age	status	email	income
9	Amina L.	Female	74	Married	amina@ya...	120 000
10	Alan D.	Male	18	Not_mar	alan@gmai...	35 000
13	Alfi A.	Male	67	Married	alfi@gmail...	88 000

For sure, such type of human oriented interfaces can be very useful for all companies that face the problem of efficient target selection of clients.

An Issue of Hedges

There is one thing that disorients me in our Direct Marketing System. Using Zadeh's definition of the *very* intensifier it follows that the curve for *very young*, must hit the values 0 and 1 at exactly the same places as the curve for *young*. It is counterintuitive in this particular application (as well as others), since it can be absolutely possible that someone is *young* without it being absolutely true that he is *very young*. This contradiction, no doubts, gets even worse with *very very young*, *very very very young*, etc. According to Zadeh's, they all hit the values 1 and 0 at the same place as *young*.

A different model for the hedges may likely be necessary for a future improvement, that narrows down the range of '*very young*' whose membership values are 1 when comparing with that of '*young*' for example.

Conclusion

The main goal of this research was to demonstrate the effectiveness of Computing with Words approach in natural query processing. In a nutshell, it allows us to form queries in natural language, which is impossible using a standard query mechanism, thus simplifying the life for an expert.

In certain areas, like Direct Marketing, target selection of information from databases has very blurred conditions. Fuzzy queries can be very efficient there. Similarly, fuzzy queries can be used in the variety of other fields. Namely, in selecting tourist services, real estate, etc.

To conclude, the use of natural language in decision problems is highly beneficial when the values cannot be expressed by means of numerical values. That happens quite often, since in natural language, truth is a matter of degree, not an absolute.

There are future improvements. However, those are mostly some minor technicality such as the matter of linguistic hedges being counterintuitive and some auxiliary, cosmetic functionality such as a parser and GUI when considering some system development.

References

(Herrera and Herrera-Viedma, 2000) F. Herrera, E. Herrera-Viedma. Linguistic decision analysis: steps for solving decision problems under linguistic information. Fuzzy Sets and Systems 115 (2000)

(Herrera, et al., 2009) F. Herrera, S. Alonso, F. Chiclana, E. Herrera-Viedma. Computing with words in decision making: foundations, trends and prospects. Fuzzy Optim Decis Making (2009) 8:337–364 DOI 10.1007/s10700-009-9065-2

(Mederia and Sousa, 2002) S. Maderia, Joao M. Sousa. Comparison of Target Selection Methods in Direct Marketing. Eunite, 2002

(Martinez, et al., 2010) L. Martinez, D. Ruan, F. Herrera. Computing with Words in Decision support Systems: An overview on Models and Applications. International Journal of Computational Intelligence Systems, Vol.3, No.4, 382-395, 2010.

(Ribeiro and Moreira, 2003). R. A. Ribeiro, Ana M. Moreira. Fuzzy Query Interface for a Business Database. International Journal of Human-Computer Studies, Vol 58 (2003) 363-391.

(Ying, 2002) M. Ying. A formal model of computing with words. IEEE Transactions on Fuzzy Systems, 10(5):640–652, 2002.

(Zadeh, 1965) L. A. Zadeh. Fuzzy sets. Information and Control, 8:338–353, 1965.

(Zadeh, 1975) L. A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. Part i. Information Sciences, 8(3):199–249, 197

(Zadeh, 1996) L. A. Zadeh. Fuzzy Logic = Computing with Words. Life Fellow, IEEE Transactions on Fuzzy Systems, Vol.4,1996.

(Zadeh, 2002) L. A. Zadeh. From Computing with Numbers to Computing with Words – from Manipulation of Measurements to Manipulation of Perceptions. Int. J. Appl. Math. Comput. Sci., Vol. 12, No. 3, 307-324, 2002.

Appendix I

Sample table data

id	name	gender	age	status	email	revenue
1	Pakita S.	Female	22	Married	pakita883@...	105 000
2	Tom M.	Male	23	Married	tom@gmail...	120 500
3	Akbota S.	Female	25	Not_married	akbota@gm...	70 000
4	Iliyas T.	Male	28	Not_married	iliyas@gma...	305 000
5	Ernar M.	Male	32	Married	era@gmail....	890 009
6	Kaminari S.	Female	40	Married	kaminari@...	55 000
7	Rus K.	Male	24	Not_married	rus_kamun...	200 000
8	Karl L.	Male	50	Married	karl@hotm...	200 300
9	Amina L.	Female	74	Married	amina@ya...	120 000
10	Alan D.	Male	18	Not_married	alan@gmai...	35 000
11	Madina D.	Female	34	Not_married	madina_@...	30 000
12	Adam S.	Male	58	Married	adam@gm...	42 000
13	Alfi A.	Male	67	Married	alfi@gmail...	88 000
14	Farida D.	Female	53	Not_married	far@mail.c...	164 000
15	Meir A.	Male	23	Not_married	meir@g.....	133 000