Zadeh's Vision of Going from Fuzzy to Computing With Words: from the Idea's Origin to Current Successes to Remaining Challenges

Vladik Kreinovich
Dept. of Computer Science, University of Texas at El Paso
El Paso, TX 79968, USA, vladik@utep.edu

Need for incorporating expert knowledge into automated systems. In order to better understand Zadeh's vision of going from fuzzy to computing with words, let us recall the reason why we need to use words in computations and whey fuzzy techniques were invented in the first place. These techniques come from the need for expert knowledge.

In many application areas, we rely on experts: medical doctors diagnose and cure diseases, civil engineers design reliable bridges and roads, pilots control planes, etc.

Some experts are more skilled than others. Every patient wants to be diagnosed and treated by the best doctors, every passenger wants to be flown by the best pilots, etc. The problem is that realistically, the best doctor cannot physically treat all the millions of illnesses, the best pilot cannot him- or herself fly all thousands of planes.

Since we cannot have the best medical doctor treat every patient, we should help all the other doctors use the knowledge and experience of the best ones – and maybe even allow the patients themselves to use this knowledge. Similarly, we should help all the pilots to use the knowledge and experience of the best ones. In general, we want all the experts to use the expertise of the best ones. For that, we need to incorporate the knowledge of the best experts into an automated system that others can use.

Expert knowledge is often imprecise ("fuzzy"): need to use (imprecise) words in computations. At first glance, incorporating expert knowledge into an automated system sounds straightforward: we can ask the experts how they make their decisions and incorporate the corresponding rules into the automated system. This works when experts can describe their decisions in precise terms, with rules like "if a flu patient has a fever above 101 F, the patient should take advil". In practice, however, experts often describe their knowledge by using imprecise ("fuzzy") words from natural language. For example, an expert driver, when describing his driving strategy, does not say "if a car is at a distance of 20 feet in front starts breaking, I hit the breaks for 0.4 second with a force of 3.6 Newtons". Instead, a driver will say something like "If a fast car in front is close and starts breaking, I hit the breaks hard". Similarly, a medical doctor does not express his decision on when to perform a certain type of surgery on a tumor by specifying the tumor's exact size. Instead, the doctor uses imprecise rules with conditions like "If the tumor is small".

First Zadeh's idea: fuzzy techniques as a way to deal with imprecise ("fuzzy") words. Fuzzy logic [4] is a technique for transforming imprecise expert rules into precise decision, precise control, etc. The main idea behind fuzzy logic is as follows: since we are not sure whether a given value x is, e.g., small, we assign a *degree* of smallness to different values x.

How can we represent these degrees inside a computer? In the computer, everything is represented as 0s and 1s. For example, "true" is usually represented as 1 and "false" as 0. We want degrees *intermediate* between 0 and 1, so it is natural to use numbers from the interval $[0, 1]$.

How to elicit fuzzy degrees. Which numbers from the interval $[0, 1]$ should we choose? There are many reasonable ways to elicit these numbers: We can poll experts and take the fraction of those believes x to be the small as the desired degree. We can ask an expert to mark his/her degree of degree on a Likert scale, e.g., from 0 to 5, and then, if the expert marks 3, take 3/5 as the desired degree, etc.

The idea becomes more complex: need to combine fuzzy degrees. At first glance, the idea of fuzzy techniques is very simple and straightforward. However, it gets more complex if we take into account that in many practical situations, in order to make an appropriate decision, we need to find the degree of truth in a complex statement like "a car in front is close *and* going fast". Ideally, to get such degrees, we should ask the expert about all possible combinations of distance and speed. However, there are too many possible combinations, so it is not realistically possible to ask the expert about all of them.

As a result, we need to estimate the degrees $d(A \,\&\, B)$ of complex statements like $A \,\&\, B$ by using only the known degrees of truth $d(A)$ and $d(B)$ of the component statements A and B. In other words, we need an algorithm $f_\&$ that transforms the degrees $d(A)$ and $d(B)$ into an estimate $f_\&(d(A), d(B))$ for $d(A \,\&\, B)$. Which algorithms $f_\&$ should we choose? Since $A \,\&\, B \equiv B \,\&\, A$ and $A \,\&\, (B \,\&\, C) \equiv (A \,\&\, B) \,\&\, C$, it is reasonable to select algorithms for which the resulting estimates coincide, i.e., for which $f_\&(a, b) = f_\&(b, a)$ and $f_\&(a, f_\&(b, c)) = f_\&(f_\&(a, b), c)$. Algorithms that satisfy these properties (and several other reasonable properties) are known as *t-norms*. Similar properties of an "or"-operation lead to the notion of a *t-conorm*, etc.

Which t-norm and t-conorm should we select: an additional complexity. There exist many different t-norms and t-conorms. Different t-norms and t-conorms lead, in general, to different recommendations. Which of these should we choose? At present, t-norms are selected *empirically* (if selected at all :-), so that for many complex statements, the elicited degree $d(A\,\&\,B)$ is, on average, the closest to the estimate $f_\&(d(A), d(B))$.

This empirical selection was first implemented for the MYCIN medical expert system [1]. The authors of the corresponding empirical study hoped that the resulting t-norm is a general description of human reasoning. Alas, when they applied their idea to geophysics, it turned out that the medically best t-norm (MYCIN) is not appropriate for geophysics at all. After the fact, it makes sense: e.g., in search for oil, it makes sense to start drilling a well once there is a reasonable expectation that this well will be productive – and it is OK that a large portion of these wells do not produce, as long as on average, we are successful. In contrast, in medicine, we do not want to perform a serious surgery on a patient unless we are absolutely sure about the diagnosis. In short, in medicine, experts use very conservative estimates, while in geophysics, they use more optimistic ones.

Yet more complexity emerges when we move from traditional fuzzy logic to more adequate implementations of computing with words. As we have just mentioned, the result of applying fuzzy techniques changes if we select a different t-norm, i.e., a different way to combine fuzzy degrees. Even with the same t-norm, the result depends on the exact values of these degrees. This dependence needs to be taken into account since, in general, for the same statement, different experts produce slightly different degrees.

Traditional fuzzy logic uses one of these degrees – or, e.g., their average. But even if we select an expert, we still have an uncertainty because usually, an expert is not sure about his or her degree of belief in a statement. Indeed, an expert may comfortably select 4 (rather than 3 or 5) on a 0 to 5 scale, so we estimate her degree of belief as $4/5 = 0.8$ (and not $3/5 = 0.6$ or $5/5 = 1.0$). However, by using this scale, we only get 6 possible degrees of belief: 0. 0.2, …, 1.0. If we want a more accurate description of the expert's degree of belief, we need to use a larger scale, e.g., 0 to 100. However, an expert usually cannot meaningfully distinguish between 81 and 82 on this scale.

A more adequate representation of expert uncertainty is that instead of a single degree value, we use the *range* of possible values of degree; see, e.g., [2]. For example, if an expert is comfortable with values from 80 to 85, we use the whole interval $[0.8, 0.85]$ as the description of the expert's degree of belief. An expert may also feel more comfortable with some of these possible values and less with others. To describe this, we can assign, to each value d from the corresponding interval $[\underline{d}, \overline{d}]$, a degree $\mu(d)$ to which this expert is comfortable with this value d. In this *type-2* approach, for each x, the degree is not a number and not an interval, but a fuzzy set; see, e.g., [2].

Zadeh's vision of computing with words. We see that the original simple idea of Lotfi Zadeh has led to very complex implementations, with complex math, and many remaining open problems. Every time we try to solve one of these problems, we make it even more complex. Maybe it is time to rethink the whole approach? We definitely need to use words, and it is reasonable to use words in the conclusions, e.g., to conclude that the patient most probably has a flu. In other words, we need to transform words into words. How this is done now:

- we start with words from natural language;
- we transform them into numbers (intervals, etc.);
- we process these numbers; and
- we transform the resulting number into a natural language word describing the conclusion.

Why we use numbers? Only because we know how to process numbers. Zadeh's new idea [3] is to cut the middleman:

- start with words,
- process words,
- produce the words as a result.

In other words, his idea is to compute with words instead of computing with numbers.

Zadeh's vision of computing with words: challenge. Ideally, we should operate directly with words. For example, we should be able to add small and medium and get – what? This is the gist of numerous Zadeh's examples like

- most Swedes are tall,
- Johannes is a Swede,
- what is the probability that Johannes is very tall?

The main challenge is that we are still far from this vision!

References.

[1] B. G. Buchanan and E. H. Shortliffe, *Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, Addison-Wesley, Reading, MA, 1984.

[2] J. M. Mendel and D. Wu, *Perceptual Computing: Aiding People in Making Subjective Judgments*, John Wiley and IEEE Press, 2010.

[3] L. A. Zadeh, "From computing with numbers to computing with words: from manipulation of measurements to manipulation of perceptions", *IEEE Transactions on Circuits and Systems I*, 1999, Vol. 45, No. 1, pp. 105–119.

[4] L. A. Zadeh, "Fuzzy sets", *Information and Control*, 1965, Vol. 8, pp. 338–353.