3-2020

Towards Making Fuzzy Techniques More Adequate for Combining Knowledge of Several Experts

Nguyen Hoang Phuong
Thang Long University, nhphuong2008@gmail.com

Vladik Kreinovich
The University of Texas at El Paso, vladik@utep.edu

Towards Making Fuzzy Techniques More Adequate for Combining Knowledge of Several Experts

Hoang Phuong Nguyen[1] and Vladik Kreinovich[2]
[1]Division Informatics, Math-Informatics Faculty
Thang Long University
Nghiem Xuan Yem Road
Hoang Mai District
Hanoi, Vietnam, nhphuong2008@gmail.com
[2]Department of Computer Science
University of Texas at El Paso
500 W. University
El Paso, TX 79968, USA, vladik@utep.edu

Abstract

In medical and other applications, expert often use rules with several conditions, each of which involve a quantity within the domain of expertise of a different expert. In such situations, to estimate the degree of confidence that all these conditions are satisfied, we need to combine opinions of several experts – i.e., in fuzzy techniques, combine membership functions corresponding to different experts. In each area of expertise, different experts may have somewhat different membership functions describing the same natural-language ("fuzzy") term like small. It is desirable to present the user with all possible conclusions corresponding to all these membership functions. In general, even if, for each area of expertise, we have only a 1-parametric family characterizing different membership function, then for rules with 3 conditions, we already have a difficult-to-interpret 3-parametric family of possible consequences. It is thus desirable to limit ourselves to the cases when the resulting family is still manageable – e.g., is 1-parametric. In this paper, we provide a full description of all such families. Interestingly, it turns out that such families are possible only if we allow non-normalized membership functions, i.e., functions for which the maximum may be smaller than 1. We argue that this is a way to go, since normalization loses some information that we receive from the experts.

1 Formulation of the Problem

Need for fuzzy techniques. In many practical problems, people make decisions:

- medical doctors treat patients,

- pilots control planes, etc.

In all these cases, some people perform better than others. It is therefore desirable to incorporate the knowledge of these top performers into a computer-based system that would provide an advice to others – or even, if appropriate, replace human controllers and human decision makers altogether.

In most cases, top performers are perfectly willing to share their knowledge and their experience. The challenge is that often, this knowledge and this experience comes in the form of imprecise ("fuzzy") words from natural language, such as "small", "high", etc. For example, a medical doctor may give recommendations what to do is a patient has *high* fever or *strong* cough or *small-size* rash, without specifying what exactly these words mean.

We humans more or less understand knowledge presented in these terms, but computers, in a nutshell, only understand numbers. It is therefore desirable to translate natural-language rules into numbers.

This was, in effect, the main purpose of fuzzy techniques invented by Lotfi Zadeh in the 1960s; see, e.g., [1, 2, 3, 5, 6, 7].

Fuzzy logic: main idea. To describe a degree to which an expert is confident in his or her statement, we can ask the expert to make his/her degree of confidence, e.g., on a scale from 0 to 10, so that:

- 0 means no confidence at all,

- 10 means full confidence, and

- intermediate degrees correspond to intermediate confidence values.

The resulting number depends on what scale we use, whether it is on a scale from 0 to 5 or from 0 to 10, etc. To get an estimate which is not depending on the scale, we can divide the corresponding number by the scale size. So, e.g., if an expert marks 7 on a 0 to 10 scale, we describe his/her degree of confidence by the number $7/10 = 0.7$. This way, no matter what the scale is:

- the degree 0 corresponds to no confidence at all,

- the degree 1 corresponds to full confidence, and

- values intermediate between 0 and 1 corresponding to partial confidence.

To describe a natural-language property P (e.g., "small"), we ask the expert to describe, for each value x of the corresponding quantity, the expert's degree of confidence that this values satisfies the corresponding property (e.g., that a

given value x is small). The corresponding mapping that assigns, to each number x, a degree to which the property holds for this value, is called a *membership function*. The corresponding degree is denoted by $\mu_P(x)$.

Of course, there are infinitely many real numbers, so an expert cannot provide us with the confidence degree $\mu_P(x)$ for all these numbers. What we normally do is:

- to ask the expert to estimate this degree for some of these values $x_1, \ldots, x_n$, and then

- to interpolate and extrapolate to other values x.

The more values n we ask about, the more adequately we describe the expert's knowledge.

How can we interpolate/extrapolate? A natural way to perform this interpolation/extrapolation is:

- to elicit these values from several experts,

- to get a good idea of what types of dependencies on x – i.e., to come up with a parametric family of functions $\mu_P(c, x)$ that describes the opinions of different experts for different values of the parameter(s) c, and then

- for each individual expert, to find the value(s) c for which his/her estimates d_i best fit this family, i.e., for which $\mu_P(c, x_i) \approx d_i$ for all i.

Need for "and"-operations. In many applications, an expert checks *several* conditions before making a decision. For example, a medical doctor can make a certain recommendation if the fever is high *and* the blood pressure is high. In general, these conditions have the form $P(x) \& Q(y)$ for some properties P and Q. We may even have not just two, but three or more conditions combined.

To describe such conditions in numerical terms, we need to know, for each pair (x, y), what is the expert's degree of confidence that x has the property P *and* that y has property Q.

We already mentioned that even when we have only one quantity x, we need to ask many (n) questions to the expert. To get an adequate description of the expert's knowledge, we need to make this n is large as possible – i.e., this n should be close to a limit of how many questions a person can answer in reasonable time. The problem is that if we consider n possible values of x and n possible values of y, then we get n^2 possible pairs. The value n was already at the limit. So, since $n^2 \gg n$, there is no way that we can ask the expert about all possible pairs. And if we have a rule that uses three conditions, then we need to consider all n^3 possible triples, and the situation will be even worse.

Since we cannot directly elicit the degree of composite statements like $P(x) \& Q(y)$ from the expert, the only thing we can do is estimate these degree based on the known estimates $\mu_P(x)$ and $\mu_Q(y)$ of the expert's degree of confidence in the component statements.

For this purpose, we need an algorithm that would transform the degree of confidence a in a statement A and the degree of confidence b in a statement B into an estimate for the combined statement $A \& B$. Such an algorithm is called an *"and"-operation*, or, for historical reasons, a *t-norm*. We will denote the result of applying the "and"-operation by $f_\&(a, b)$.

This algorithm must satisfy several natural properties.

- Since $A \& B$ means the same as $B \& A$, the estimates for these two formulas should be the same: $f_\&(a, b) = f_\&(b, a)$. In mathematical terms, an "and"-operation must be *commutative*.

- Similarly, since $A \& (B \& C)$ means the same as $(A \& B) \& C$, we must have $f_\&(a, f_\&(b, c)) = f_\&(f_\&(a, b), c)$, i.e., an "and"-operation must be *associative*.

- If our degree of confidence in one (or both) of the statements increases, then we expect our degree of confidence in $A \& B$ to also increase (or at least to remain the same). Thus, the function $f_\&(a, b)$ must be monotonic in both variables: if $a \le a'$ and $b \le b'$, then $f_\&(a, b) \le f_\&(a', b')$.

- If we change the degrees of confidence a little bit, we expect that the estimate also not change much. In other words, we should require the function $f_\&(a, b)$ to be continuous.

- If A is absolutely true (i.e., if $a = 1$), then $A \& B$ is true if and only if B is true. So, our degree of confidence $f_\&(1, b)$ should be equal to our degree of confidence b in the statement B: $f_\&(1, b) = b$.

- Similarly, if A is absolutely false (i.e., if $a = 0$), then $A \& B$ should be false: $f_\&(0, b) = 0$.

There is a known classification of all the "and"-operations that satisfy all these properties; see, e.g., the above-mentioned fuzzy textbooks.

Case when the same expert provides all the estimates. In some cases, the same expert can meaningfully evaluate all the conditions.

Different experts may provide different membership functions. If we use k parameters for describing this difference, we end up with a k-parametric family of resulting decisions. For small k, this is a reasonable-size set of recommendations between them a user of the corresponding computer-based system can choose.

For example:

- some experts may prefer a more aggressive attitude towards the disease, recommending higher doses of medicine, while

- other medical doctors prefer smaller doses (at least at first), trying to minimize negative side effects of the corresponding medicine.

4

Often, we need to combine the knowledge of several experts. The above scheme works well when we have a single expert.

A challenging problem emerges since in many practical situations, decisions are made based on the opinions of *several* experts. For example, in medicine, many decisions – and many rules – incorporate opinions of several experts. For example, if a patient has a persistent cough, he may be examined:

- by his regular doctor,

- by a specialist in allergies, and

- by a specialist in lung diseases, etc.

The natural-language rules may be fix, this rule may be taken from the well-approved recommendations and/or textbooks, but the translations of the different terms from this rule will be given by *different* experts.

If the rule includes an opinion of three experts, then, taking into account that there is a k-parametric variety of membership functions corresponding to each expert, we will have a $(3k)$-parametric family of resulting recommendations. Indeed, to describe a specific recommendation, we will need to list:

- k parameters corresponding to the first expert,

- k parameters corresponding to the second expert, and

- k parameters corresponding to the third expert.

Even for $k = 1$, we get too big a family of recommendations, too big to take into account or even to describe graphically – graphically we can only describe 2-D sets reasonably well.

What we do in this paper. In this paper, we analyze this challenging problem and, as a result of this analysis, come up with some ideas on how to solve it – i.e., how to make fuzzy techniques more adequate for combining knowledge of several experts.

2   Analysis of the Problem and the Resulting Idea

Towards precise formulation of the problem. Let us start our analysis with the simplest case of the above problem, when we have a single combination $A(x) \& B(y)$ if two conditions. To estimate the degree to which this combination is true, we ask two experts:

- an expert specializing in properties related to the quantity x, and

- an expert specializing in properties related to the quantity y.

For each area of expertise, we have several experts who may provide us with somewhat different membership functions. To keep everything feasible, we select a small number k (e.g., $k = 1$), and describe membership functions corresponding to different x-experts by a k-parametric family $\mu_A(c, x)$, where $c = (c_1, \ldots, c_k)$ is the tuple of the corresponding parameters.

Similarly, we describe membership functions corresponding to different y-experts by a k-parametric family $\mu_B(c', y)$, where c' is also a k-dimensional tuple.

In general, if we combine these two degree, we get an expression $f_\&(\mu_A(c, x), \mu_B(c', y))$ that depends on $2k$ different parameters:

- k parameters forming a tuple c and

- k parameters forming a tuple c'.

We would like to make sure that all these functions can be described by only k parameters.

We assume that the "and"-operation is fixed. In this case, the only choice we can make is selecting different families $\mu_A(c, x)$ and $\mu_B(c', y)$. So, we arrive at the following precise formulation of the problem.

Precise formulation of the problem. For a given "and"-operation $f_\&(a, b)$, describe the conditions on the k-parametric families $\mu_A(c, x)$ and $\mu_B(c', y)$ for which the class of all possible functions of the type $f_\&(\mu_A(c, x), \mu_B(c', y))$ is also k-parametric.

Discussion. The "and"-operation can be very complicated. So, let us first consider one of the simplest operations $f_\&(a, b) = a \cdot b$ – which, by the way, was one of the operations proposed by Zadeh himself in his pioneering paper on fuzzy techniques.

For this simple "and"-operation, the above problem takes the following form.

Particular case of this problem when $f_\&(a, b) = a \cdot b$. Describe the conditions on the k-parametric families $\mu_A(c, x)$ and $\mu_B(c', y)$ for which the class of all possible functions of the type $\mu_A(c, x) \cdot \mu_B(c', y)$ is also k-parametric.

The general case can be reduced to this special case. Let us show that the general case of the above problem can be reduced to the case $f_\&(a, b) = a \cdot b$.

Indeed, according to [4], every "and"-operation, for any $\varepsilon > 0$, can be approximated, with accuracy ε, by an "and"-operation of the type $f^{-1}(f(a) \cdot f(b))$ for some monotonic function $f(a)$; here, $f^{-1}(a)$ means an inverse function, i.e., a function for which $f^{-1}(f(a)) = a$ for all a. Since ε can be arbitrarily small, from the practical viewpoint, this means that we can safely assume that the original "and"-operation has this form.

For this "and"-operation, the desired property is that the family $f^{-1}(f(\mu_A(c, x)) \cdot f(\mu(c', y)))$ depends only on k parameters. Since the functions $f(a)$ and $f^{-1}(a)$ are invertible, they do not change the dimension of (= number of parameters in) the corresponding family. Thus, the desired property is

equivalent to saying that the family of functions $f(\mu_A(c, x)) \cdot f(\mu(c', y))$ depends only on k parameters.

If we denote $\nu_A(c, x) \stackrel{\text{def}}{=} f(\mu_A(c, x))$ and $\nu_B(c', y) \stackrel{\text{def}}{=} f(\mu_B(c', y))$, then the question takes the following form: Describe the conditions on the k-parametric families $\nu_A(c, x)$ and $\nu_B(c', y)$ for which the class of all possible functions of the type $\nu_A(c, x) \cdot \nu_B(c', y)$ is also k-parametric.

This is exactly the problem corresponding to the case when $f_\&(a, b) = a \cdot b$. So, if we can solve this problem, then from its solutions $\nu_A(c, x)$ and $\nu_B(c', y)$, we can determine the original membership functions as $\mu_A(c, x) = f^{-1}(\nu_A(c, x))$ and $\mu_B(c', y) = f^{-1}(\nu_B(c', y))$.

In view of this reduction of the general case to the case when $f_\&(a, b) = a \cdot b$, in the following text, we will only consider this special case, when the "and"-operation is simply a product.

Let us further analyze the problem. Let us fix some tuples $c^{(0)}$ and $c'^{(0)}$ in the interior of a region of all possible values of these tuples. Then, if we take the values c in a small vicinity of the fixed tuple $c^{(0)}$, the functions

$$\mu_A(c, x) \cdot \mu_B\left(c'^{(0)}, y\right) \tag{1}$$

will already form a k-parametric family. Thus, for each tuple c' in the vicinity of $c'^{(0)}$, the product

$$\mu_A\left(c^{(0)}, x\right) \cdot \mu_B(c', y) \tag{2}$$

should also be within this family – otherwise, the dimension of the entire family (corresponding to all possible pairs (c, c')) will be larger than k.

So, each product of the type (2) should coincide with one of the products of type (1). In other words, for each tuple c', we should have a tuple $c(c')$ depending on c' for which

$$\mu_A(c(c'), x) \cdot \mu_B\left(c'^{(0)}, y\right) = \mu_A\left(c^{(0)}, x\right) \cdot \mu_B(c', y). \tag{3}$$

To simplify the resulting formula, let us divide both sides by the product $\mu_B\left(c'^{(0)}, y\right) \cdot \mu_A\left(c^{(0)}, x\right)$. This way, the left-hand side will only depend on x and the right-hand side will only depend on y:

$$\frac{\mu_A(c(c'), x)}{\mu_A\left(c^{(0)}, x\right)} = \frac{\mu_B(c', y)}{\mu_B\left(c'^{(0)}, y\right)}. \tag{4}$$

The right-hand side of this equality does not depend on x; thus, the left-hand side does not depend on x either – so it must depend only on c. If we denote this ratio by $r(c)$, then we conclude that

$$\frac{\mu_A(c, x)}{\mu_A\left(c^{(0)}, x\right)} = r(c), \tag{5}$$

i.e., that

$$\mu_A(c, x) = r(c) \cdot \mu_A\left(c^{(0)}, x\right). \tag{6}$$

So, all the functions $\mu_A(c, x)$ from the family can be obtained from one of them by multiplying by an appropriate constant. Thus, in effect, all the functions from this family have the form $C \cdot \mu_A(x)$ for some function $\mu_A(x)$ and for different possible values C.

By the way, since different functions from this family differ only by one parameter – the multiplicative constant – this family is 1-dimensional, i.e., we have $k = 1$.

Similarly, the left-hand side of the equality (4) does not depend on y; thus, the right-hand side does not depend on y either – so it must depend only on c'. If we denote this ratio by $r'(c')$, then we conclude that

$$\frac{\mu_B(c', y)}{\mu_B\left(c'^{(0)}, y\right)} = r'(c'), \tag{7}$$

i.e., that

$$\mu_B(c', y) = r'(c') \cdot \mu_B\left(c'^{(0)}, y\right). \tag{8}$$

So, all the functions $\mu_B(c', y)$ from the family can be obtained from one of them by multiplying by an appropriate constant. Thus, in effect, all the functions from this family have the form $C' \cdot \mu_B(y)$ for some function $\mu_B(y)$ and for different possible values C'.

We have shown that if the above properties is satisfied, then the corresponding families have this form. One can easily show that, vice versa, if we have families of the type $\mu_A(C, x) = C \cdot \mu_A(x)$ and $\mu_B(C', y) = C' \cdot \mu_B(y)$, then the products $\mu_A(C, x) \cdot \mu_B(C', y)$ have the form $(C \cdot C') \cdot \mu_A(x) \cdot \mu_B(y)$ and thus, depend only on a single parameter $C \cdot C'$.

So, for the case when $f_\&(a, b) = a \cdot b$, we have a general solution to the above problem.

Solution to the main problem when $f_\&(a, b) = a \cdot b$. The following two conditions are equivalent to each other:

- For two k-parametric families $\mu_A(c, x)$ and $\mu_B(c', y)$, the class of all possible functions of the type $\mu_A(c, x) \cdot \mu_B(c', y)$ is also k-parametric.

- $k = 1$ and both families can re-parameterized into families $C \cdot \mu_A(x)$ and $C' \cdot \mu_B(y)$, where $\mu_A(x)$ and $\mu_B(y)$ are fixed and C and C' are arbitrary real numbers.

Discussion. By using the above reduction, we can formulate the following general solution for the above problem.

Solution to the main problem: general case. Let

$$f_\&(a, b) = f^{-1}(f(a) \cdot f(b)).$$

Then, the following two conditions are equivalent to each other:

8

- For two k-parametric families $\mu_A(c, x)$ and $\mu_B(c', y)$, the class of all possible functions of the type $f_\&(\mu_A(c, x), \mu_B(c', y))$ is also k-parametric.

- $k = 1$ and both families can re-parameterized into families $\mu_A(C, x) = f^{-1}(C \cdot \nu_A(x))$ and $\mu_B(C', y) = f^{-1}(C' \cdot \nu_B(y))$, where $\nu_A(x)$ and $\nu_B(y)$ are fixed and C and C' are arbitrary real numbers.

Discussion: so maybe we do not need normalization. When we briefly described the traditional fuzzy technique, we skipped one important step that this technique normally includes: the *normalization* step.

Namely, for each property P, usually, after we elicit, for different values x, the degrees $\mu_P(x)$, and if it turns out that $m \stackrel{\text{def}}{=} \max_x \mu_P(x) < 1$, we "normalize" this membership function by dividing all its values by m, i.e., by taking the values $\nu_P(x) = \dfrac{\mu_P(x)}{m}$. For such normalized membership functions, we always have $\max_x \nu_P(x) = 1$.

For some properties like "small", normalization is not necessary: there is a value $x = 0$ which is clearly absolutely small. However, for other properties like "medium", we may not have a value which the expert would consider to be absolutely medium – so, in the normal fuzzy methodology, we would need to apply normalization.

It all may sound reasonable until we realize that if a function $\mu_A(x)$ is normalized, then no other function of the type $C \cdot \mu_A(x)$ will be normalized: indeed, if $\max_X \mu_A(x) = 1$, then $\max_x (C \cdot \mu_A(x)) = C$ and is, thus, normalized only if $C = 1$.

So, if we limit ourselves to normalized membership functions, the desired families cannot exist: each of them would consist of only one membership function, which defeats the purpose of describing difference between the experts. This is a rather gloomy conclusion, but we can turn it around and make it positive: *if we want to describe the difference in opinion between different experts, then we need to allow non-normalized membership functions.*

Normalization only loses information. There is another, simpler reason why we want to skip the normalization step:

- If we do not perform the normalization, then each degree $\mu_P(x)$ describes the original expert's opinion.

- However, when we apply normalization, we lose some of the information provided by the expert: namely, the information about the largest of the expert's confidence degree.

For example, if we apply normalization, we are no longer be able to distinguish between the two clearly different cases:

- the case when the expert considered, e.g., the value $x = 2$ as absolutely medium, so that all rules with medium size as a condition are absolutely applicable to this value, and

- the case when the expert is not 100% sure that $x = 2$ is medium, and is, thus, not absolutely sure that rules with medium size as a condition are applicable to this value.

Acknowledgments

References

[1] R. Belohlavek, J. W. Dauben, and G. J. Klir, *Fuzzy Logic and Mathematics: A Historical Perspective*, Oxford University Press, New York, 2017.

[2] G. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, Upper Saddle River, New Jersey, 1995.

[3] J. M. Mendel, *Uncertain Rule-Based Fuzzy Systems: Introduction and New Directions*, Springer, Cham, Switzerland, 2017.

[4] H. T. Nguyen, V. Kreinovich, and P. Wojciechowski, "Strict Archimedean t-norms and t-conorms as universal approximators", *International Journal of Approximate Reasoning*, 1998, Vol. 18, Nos. 3–4, pp. 239–249.

[5] H. T. Nguyen, C. L. Walker, and E. A. Walker, *A First Course in Fuzzy Logic*, Chapman and Hall/CRC, Boca Raton, Florida, 2019.

[6] V. Novák, I. Perfilieva, and J. Močkoř, *Mathematical Principles of Fuzzy Logic*, Kluwer, Boston, Dordrecht, 1999.

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