A Novel Fuzzy Modifier Interpolation Rule for Computing with Words

Tanvi Dadu, Swati Aggarwal* and Nisha Aggarwal
E-mail: tanvid.co.16@nsit.net.in, swati1178@gmail.com, nishaaggarwal2810@gmail.com
Netaji Subhas University of Technology, Dwarka, New Delhi 110078, India

Keywords: type-1 fuzzy sets, interval type 2 fuzzy sets, fuzzy modifiers, interpolation rule, computing with words

Received: June 15, 2021

Computing with words is a concept that is used to solve problems with input in natural language. Modifiers are transformation functions with predefined labels used extensively in decision-making to specify the desired value of a linguistic variable defined by fuzzy sets.
In past years, few efforts have been made to study the application of Computing with Words (CW) in many domains ranging from fraud detection systems to diagnosis systems in medicine. However, the application of CW in these fields with modified Fuzzy sets did not give satisfactory results. When applied to modified Fuzzy sets, the existing interpolation rule does not cover the extreme left and extreme right-shifted fuzzy sets. Hence, there is a need to introduce a new interpolation rule when working with modifiers. This paper introduces a new Fuzzy Modifier Interpolation Rule to Type-1 Fuzzy sets and Interval Type-2 (IT-2) Fuzzy Sets to enhance the quality of results obtained when modifiers are applied.

Povzetek: V prispevku je predstavljeno novo interpolacijsko pravilo za mehke modifikatorje v mehkih nizih tipa 1 in intervalnega tipa 2.

1 Introduction

Computing with Words (CW) is a methodology that allows using words in place of numbers for computing and reasoning. It was introduced as an extension to Fuzzy Logic Systems by Zadeh [1][2][3]. It is a system of computation that offers the capability to compute with information present in a natural language. The advancement in CW has allowed a certain degree of fuzziness to the input and the propositions present in the database of the CW inference engine. The propositions in CW are specified using linguistic variables [4][5][6], whose values are words in natural language. CW engine consists of Rule Base, Fuzzy Inference Engine, and Output Generator. The IF-THEN rules in the CW engine are specified using natural language, which is modeled using either Type - 1 or Type - 2 Fuzzy sets (Interval Type-2 and General Type-2) [7].

Computing with Words (CW) has a wide variety of applications ranging from household appliances like fraud detection systems to biomedical instrumentation [8]. Medicine is one of the domains where the applications of Computing with Words has been recognized [8]. The uncertainty found in these applications is appropriately captured by fuzzy set theory. Therefore, in the past few years, new techniques in CW have been extensively applied in decision-making systems.

Modifiers are transformation functions with predefined labels applied to a fuzzy set defined on a linguistic variable to specify the desired value of a variable. Modifiers help us define input when it is not present in the given fuzzy sets. Modifiers like VERY, EXTREMELY, and connective like NOT, OR can be used to enhance the capability of the meaning specified by Fuzzy sets [9][10][11]. However, the direct application of the Fuzzy Interpolation rule on the modified fuzzy set gives poor results. While computing results, it does not incorporate the contribution from fuzzy sets shifted to the extreme left or extreme right on the application of modifiers as detailed in Section 4.

This paper introduces a novel Fuzzy Modifier Interpolation rule for Type-1 Fuzzy sets and Interval Type-2 Fuzzy sets to get better results and reduce errors when modifiers are applied to the fuzzy sets. In Section 2, we present recent relevant works followed by the proposed Fuzzy Modifier Interpolation rule for Type-1 and Interval Type-2 Fuzzy sets in Section 3. In Section 4, we present results of experiments performed on the UCLA dataset for heart disease [27] to show improvement in graphs or inference obtained upon application of our new Fuzzy Modifier Interpolation Rule. In Section 5, we discuss the results obtained for our proposed Fuzzy Modifier Interpolation rule with Fuzzy Interpolation rule. Finally, we discuss the advantages and limitations of our proposed work in Section 6 and Section 7, respectively.

2 Related works

The concept of modifier to represent linguistic hedges employing a mathematical transformation of membership functions was first introduced by Zadeh [12]. The author proposed using linguistic hedges as a power functions based operator, which applied to fuzzy sets, represents

*Corresponding author

the meaning of its operands. While applying transformations, the author used pure post modification of membership functions to represent linguistic hedges. The modifiers proposed by the author did not cover all forms of linguistic hedges. To mitigate this, another type of modifier called shifting modifiers was put forward in [13]. These modifiers are translatory modifiers that involve pure pre-modification of the membership functions.

However, traditional modifiers like powering and shifting modifiers could not handle similar categories distinguished by subtle differences. To mitigate this, De Cock and Kerre [14] introduced a new form of fuzzy modifiers, where weakening adverbs (*more or less*, *roughly*) and intensifying adverbs (*very*, *extremely*) are modeled in the inclusive and the non-inclusive interpretation. Another enhancement on powering and shifting modifiers was L-fuzzy modifiers introduced in [15] to model linguistic hedges in the L-fuzzy sets. They proposed using context through L-fuzzy relations to ensure L-fuzzy modifiers are endowed with clear inherent semantic. The proposed modifiers outperformed the traditional ones from the semantic point of view.

Since their advent, fuzzy modifiers have been used extensively in fuzzy rule-based and control systems [18, 19], analogy-Based Reasoning systems [16] and image processing or image-based retrieval systems [21]. They are used to obtain more interpretable results, limit the number of premises, dynamically modify the shape of membership functions and reduce the rule base [17, 20]. They have been used in designing databases as well, like SQLf - a well-known query language database [23, 22]. Fuzzy Modifiers have a wide range of practical applications, and therefore, it is imperative to account for the edge cases when modifiers are applied in real-world applications.

3 Approach

3.1 Preliminaries

Modifiers are a necessary part of Computing with Words. They allow us to define values in between given N Fuzzy sets. *Very*, *extremely*, *slightly*, *relatively*, *somewhat*, *quiet*, and *rather* are some of the frequently used modifiers. They are generally denoted by m.

$$X\, is\, mA \rightarrow X\, is\, f(A) \tag{1}$$

Modifier Type	Function $f(x)$
Extremely/Highly	x^3
Very/Rather	x^2
Relatively/Quite	$x^{3/2}$
Slightly	$x^{1/2}$
Somewhat	$x^{1/3}$

Table 1: Functions of The Modifiers

where f(A) is used to modify the behaviour of Fuzzy sets. The f(x) for frequently used modifiers is given in Table 1.

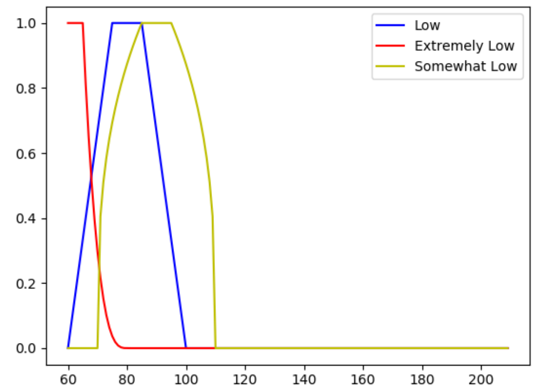


Figure 1: Modifier Applied to Type-1 Fuzzy Set

Figure 1 shows the Modifiers mentioned in Table 1 i.e. $Extremely$ and $Somewhat$ applied to Fuzzy set Low. The $f(x)$ is x^3 in case of Extremely and $x^{1/3}$ in case of Somewhat. The membership curve has shifted towards left for $extremely$ modifier, and it has shifted towards the right for $somewhat$ modifier. This is in accordance with the context of the application of modifiers.

Modifiers are extensively used in decision systems, especially in the domain of medicine. The interpolation rule for the Fuzzy set is unable to effectively deal when modifiers are applied. It cannot take contributions from the extreme left and extreme right-shifted fuzzy sets on the application of modifiers. Therefore a new Fuzzy Modifier Interpolation Rule was introduced for Type-1 Fuzzy sets.

3.2 Proposed fuzzy modifier interpolation rule on type-1 fuzzy set

In this subsection, we introduce our proposed Fuzzy Interpolation rule for Type-1 Fuzzy sets. The structure of the proposed rule consists of:

1. A set of IF-THEN rules, which is the database for inference mechanism.
2. An Antecedent with fuzzy modifier applied on it.
3. The proposed Fuzzy Modifier Interpolation rule, which consists of two cases.
   (a) **Edge Case**: It refers to the fuzzy sets having values near the boundaries of the domain.
   (b) **Non-Edge Case**: It refers to fuzzy set not having values near the boundaries of the domain.

Our proposed Fuzzy Modifier Interpolation technique predominantly deals with edge cases-Left edge case and the right edge case. Block diagram for proposed Fuzzy Modifier Interpolation rule is given in Fig 2 where m in mA_i antecedent is a modifier applied to the Fuzzy set input A_i. There are two separate cases - edge case and non-edge case, which gives Y is B as consequent.

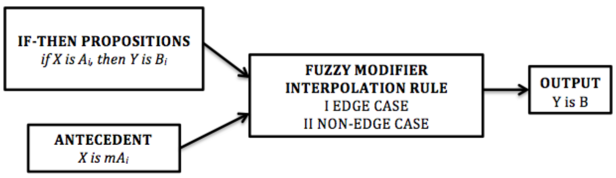


Figure 2: Block diagram for the application of Fuzzy Modifier Interpolation rule

3.2.1 Edge case

The edge case is defined as the case where the fuzzy sets have values near the boundaries of the domain. Formally, it is defined to be present when the following condition is satisfied:

$$If\ mA_i\ >\ A_i\ when\ i = n\ or\ mA_i < A_i\ when\ i = 1 \quad (2)$$

The condition (2) ensures that after application of modifiers on Type-1 fuzzy sets, the modified set is shifted either to the left extreme or right extreme.

The membership of the output Type-1 Fuzzy set obtained after application of our proposed Fuzzy Modifier Interpolation Rule is given by (3) for both left and right edge cases.

$$\mu_B(v) = f(\mu_{B_i}(v) + c) \quad (3)$$

In (3), the constant *c* defines the value of horizontal shift of fuzzy set, which is chosen after careful analysis of the domain. Figure 3 (Left). depicts the application of the proposed Fuzzy Modifier Interpolation rule for the right edge case with *relatively* modifier. Figure 3 (Right) depicts the resultant graph obtained after application of Fuzzy Modifier Interpolation rule. It shows the ability of our proposed rule to handle edge cases when fuzzy set $EXCESS$ are shifted to the right extreme and the ability to infer modified consequent fuzzy set.

3.2.2 Non edge case

This case deals where the equation (2) is not satisfied that is , $\overline{A}$ is a non-edge case fuzzy set. When a modifier m is applied, the resulting output membership is given by the following results:

$$\mu_B = sup_j(m_j \cap B_j) \quad (4)$$

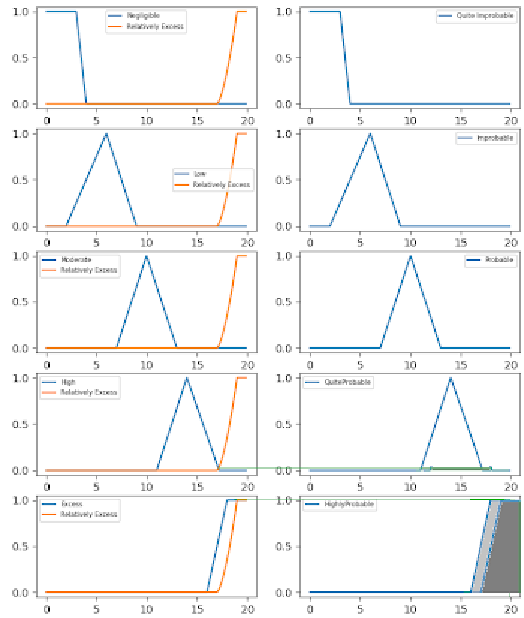


Figure 3: Fuzzy Modifier Interpolation Rule for Edge Case In Type-1 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Type-1 Fuzzy edge case
Right: Results on Application of Fuzzy Modifier Interpolation Rule

$$m_j = sup(A_j \cap A_i) \quad (5)$$

where $j = i - 1\, to\, i + 1$ and $A = mA_i$

Figure 4 (Left) depicts the application of proposed Fuzzy Modifier Interpolation rule for non edge case with extremely modifier. Figure 4 (Right) depicts the resultant graph obtained after application of Fuzzy Modifier Interpolation rule. It shows the ability of our proposed rule to deal with the non-edge case when applied to the $MODERATE$ fuzzy set. Slight changes in results are observed when compared with inferences drawn Fuzzy interpolation rule for the non-edge case.

3.3 Proposed fuzzy modifier interpolation rule on interval type-2 fuzzy set

This section extends the Fuzzy Modifier Interpolation rule to Interval Type-2 Fuzzy sets. Figure 5 shows the Modifiers mentioned in Table 1, i.e., $Extremely$ and $Somewhat$ applied to Interval Type-2 Fuzzy set 'Low'. The $f(x)$ is x^3 in case of Extremely and $x^{1/3}$ in case of Somewhat. Similar to Type-1 Fuzzy sets, the membership curve has shifted to-

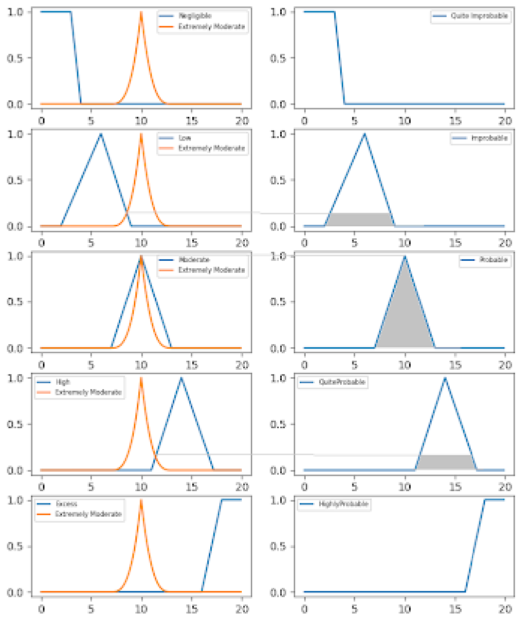


Figure 4: Fuzzy Modifier Interpolation Rule for Non-Edge Case In Type-1 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Type-1 Fuzzy non-edge case
Right: Results on Application of Fuzzy Modifier Interpolation Rule

wards the left for *extremely* modifier and towards the right for the *somewhat* modifier. This is in accordance with the context of the application of modifiers.
Similar to the Fuzzy Modifier Interpolation rule for Type-1 Fuzzy sets, the Fuzzy Modifier Interpolation rule in Interval Type-2 sets contains two cases as explained in Section 3.2:

1. Edge Case
2. Non-Edge Case

3.3.1 Edge case

Edge case is applied when (2) is satisfied by Interval Type-2 Fuzzy set $\overline{\text{A}}$. Eq (6) depicts the application of modifier to Interval Type-2 Fuzzy sets and Eq (7) gives the membership of the output edge for Interval Type-2 Fuzzy set when l^{th} rule is fired is given:

$$\mu_G^l(y) = f(\mu_G^l(y) + c) \tag{6}$$

$$\mu_B^l(y)^{edge} = \mu_G^l(y) \cap F_{edge}^l \qquad y \in Y \tag{7}$$

where F_{edge}^l is given by Eq (8) and Eq (9) :

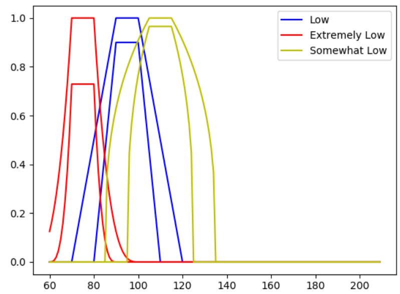


Figure 5: Modifier Applied to Interval Type-2 Fuzzy Set

$$\overline{\text{F}}_{edge}^l = sup \int_{x_1 \in X_1} \dots \int_{x_p \in X_p} [\overline{\mu}_X(x1) \star \overline{\mu}_{F^l}(x_1)]/x_1 \star \dots . \star [\overline{\mu}_X(x_p) \star \overline{\mu}_{F^l}(x_p)]/x_p \tag{8}$$

$$\underline{\text{F}}_{edge}^l = sup \int_{x_1 \in X_1} \dots \int_{x_p \in X_p} [\underline{\mu}_X(x1) \star \underline{\mu}_{F^l}(x_1)]/x_1 \star \dots . \star [\underline{\mu}_X(x_p) \star \underline{\mu}_{F^l}(x_p)]/x_p \tag{9}$$

3.3.2 Non-edge case

This case is applied when the condition (2) is not satisfied that is , $\overline{\text{A}}$ is a non-edge case. Similar to edge case for Interval Type-2 Fuzzy sets, $\mu_G^l(y)$ is modified upon application of modifier as given in Eq (6) and Eq (10) gives Membership of the output non-edge Interval Type-2 Fuzzy set when l^{th} rule is fired is given :

$$\mu_B^l(y)_{non-edge} = \mu_G^l(y) \cap F_{non-edge}^l \qquad y \in Y \tag{10}$$

where $F_{non-edge}^l$ is given by Eq (11) and Eq (12):

$$\overline{\text{F}}_{non-edge}^l = sup \int_{x_2 \in X_2} \dots \int_{x_{p-1} \in X_{p-1}} [\overline{\mu}_X(x2) \star \overline{\mu}_{F^l}(x_2)]/x_2 \star \dots . \star [\overline{\mu}_X(x_{p-1}) \star \overline{\mu}_{F^l}(x_{p-1})]/x_{p-1} \tag{11}$$

$$\underline{\text{F}}_{non-edge}^l = sup \int_{x_2 \in X_2} \dots \int_{x_{p-1} \in X_{p-1}} [\underline{\mu}_X(x2) \star \underline{\mu}_{F^l}(x_2)]/x_2 \star \dots . \star [\underline{\mu}_X(x_{p-1}) \star \underline{\mu}_{F^l}(x_{p-1})]/x_{p-1} \tag{12}$$

For l^{th} rule , The membership of the output Interval Type-2 Fuzzy set is union of membership of outputs from edge and non-edge cases in Interval Type-2 Fuzzy set as given in Eq (13).

$$\mu_B^l(y) = \mu_B^l(y)_{edge} \sqcup \mu_B^l(y)_{non-edge} \tag{13}$$

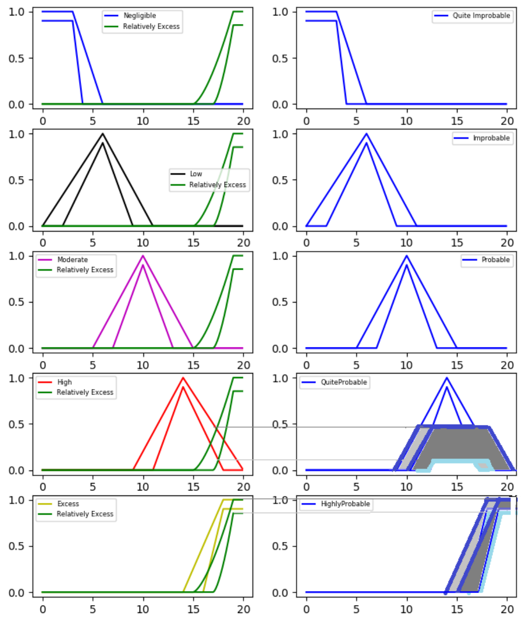


Figure 6: Example of Fuzzy Modifier Interpolation rule for edge case in Interval Type-2 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Interval Type-2 Fuzzy edge case
Right: Results obtained after application of Fuzzy Modifier Rule on Interval Type-2 Fuzzy sets for edge case

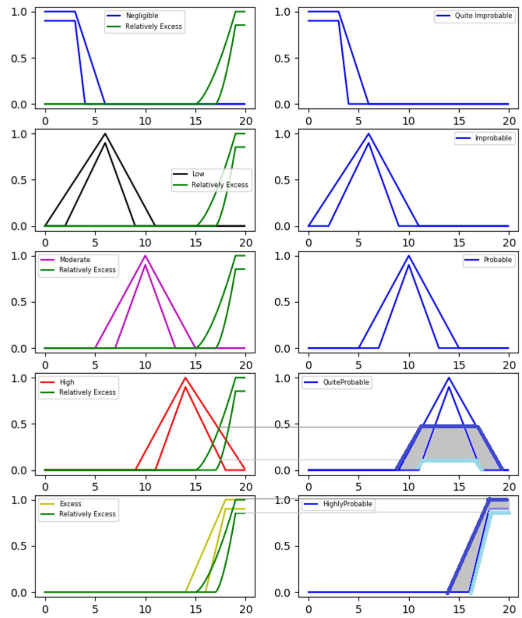


Figure 7: Example of Fuzzy Modifier Interpolation rule for non-edge case in Interval Type-2 Fuzzy sets
Left: Application of Fuzzy Modifier Interpolation Rule on Interval Type-2 Fuzzy non-edge case
Right: Results obtained after application of Fuzzy Modifier Rule on Interval Type-2 Fuzzy sets for non-edge case

Membership of the output Interval Type-2 Fuzzy set when N rules are fired is :

$$\mu_B(y) = \sqcup_{l=1}^{N} \mu_B(y) \qquad y \in Y \tag{14}$$

Figure 7 (Left) depicts an example of application of Fuzzy Modifier Interpolation Rule applied on modified Interval Type-2 Fuzzy set $EXCESS$ with $RELATIVELY$ modifier and Figure 7(Right) shows the results obtained after application of our proposed interpolation rule on Interval Type-2 Fuzzy sets for non-edge cases. Similar to Type-1 Fuzzy sets for non-edge cases, slight changes in results are observed compared with inferences drawn from the Fuzzy interpolation rule.

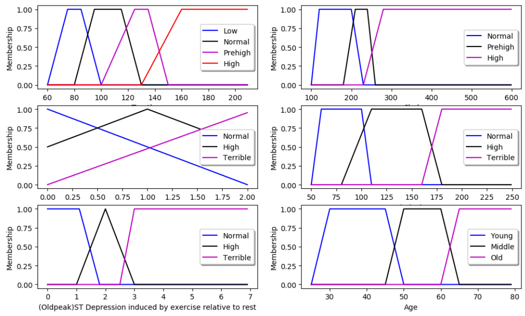
Fig 7 (Left) depicts an example of application of Fuzzy Modifier Interpolation Rule applied on modified Interval Type-2 Fuzzy set $EXCESS$ with $RELATIVELY$ modifier and Fig 7 (Right) shows the results obtained after application of our proposed interpolation rule on Interval Type-2 Fuzzy sets for edge cases. Similar to Type-1 Fuzzy sets, our proposed rule takes care of inference from fuzzy sets shifted towards extreme right and extreme left upon application of modifiers.

4 Experimentation

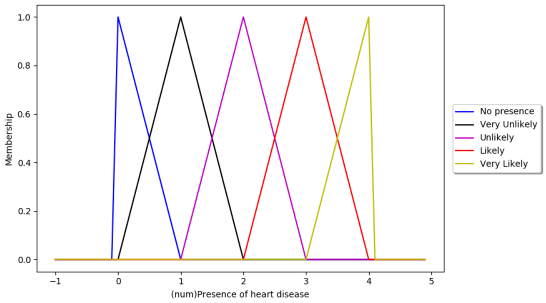
The proposed Fuzzy Modifier Interpolation for Fuzzy Sets is applied on the UCLA Cleveland dataset for heart disease for different cases. The dataset contains 76 attributes, but we are using six attributes (resting blood pressure or tresbps, serum cholesterol or chol, maximum heart rate achieved or thalach, ST depression induced by exercise relative to rest or oldpeak, resting electrocardiographic results or restecg and age) to predict the presence of heart disease.

Careful domain analysis on factors affecting heart diseases was done before formulating the selected attributes' fuzzy membership functions. Once the fuzzy sets of the selected attributes were formulated, the fuzzy rule was obtained by using Fuzzy Decision Trees Induction Method [28]. This method given by Yufei Yuan, Michael J. Shaw is based on reducing classification ambiguity with fuzzy evidence. To reduce the ambiguity while partitioning, we use a significance level of 0.45.

The experiments on Fuzzy Modifier Interpolation Rule were conducted in two phases with Phase 1 focusing on Type-1 Fuzzy sets and Phase 2 focusing on Interval Type-2 Fuzzy sets.



(a) Membership of Antecedents



(b) Membership of Consequent

Figure 8: Type-1 Fuzzy Sets for different factors affecting Heart disease

4.1 Experiments and analysis for type-1 fuzzy sets

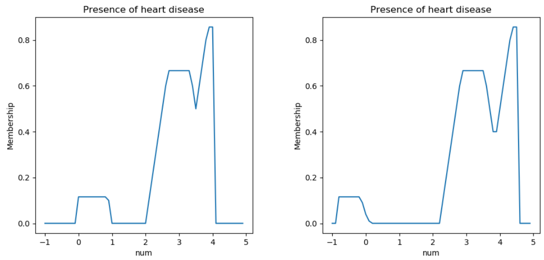
The first half of the experiment was focused on Type-1 Fuzzy sets. It was conducted for three cases - right edge case, left edge case, and non-edge cases. The antecedents and consequents are represented using Type-1 Fuzzy sets. Figure 8 (A). depicts membership of antecedents, and Figure 8 (B) depicts membership of consequent.

Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation Rule using *min T-norm* is applied to the query for right edge case and output curve obtained is depicted in Figure 9 (A) and Figure 9 (B) respectively for fuzzy interpolation rule and fuzzy modifier interpolation rule. We observe the graph obtained in Figure 9 (A) is shifted to the left and right taking care of the extreme left and right cases obtained on application of modifiers.

Input Query used for the right edge case is given in Table 2:

The obtained graphs for the edge cases were defuzzified to understand better the results obtained after applying the Fuzzy Modifier Interpolation Rule. For the left edge case, we used the least of the maximum for defuzzification. Similarly, for the right case, we used the largest of the maximum for defuzzification and centroid for the non-edge case. Figure 10 shows the comparison between the results obtained for the Fuzzy Interpolation rule and the Fuzzy Modifier Interpolation rule on defuzzification for the right edge case. In Figure 10 (B), the defuzzified values are shifted towards the right for the mean of maximum, min of maximum, and max of maximum criteria. This corroborates the ability of our proposed interpolation rule to infer from edge cases efficiently. The choice of the defuzzification technique is case-dependent and based on the shift of the fuzzy sets expected in each case.

A comparison between the Fuzzy Modifier Interpolation Rule and Fuzzy Interpolation Rule for all three cases- left edge case, non-edge case, and right edge case is shown for the Type-1 Fuzzy set. The blue curve in Figure 11 is obtained after the application of Fuzzy Interpolation Rule to Type-1 Fuzzy sets, and the Orange curve is obtained after the application of Fuzzy Modifier Interpolation Rule to Type-1 Fuzzy sets. In Figure 11 (A), The modifier is applied to the right valued Type-1 Fuzzy input, and after the application of Fuzzy Modifier Interpolation Rule, the output curve is shifted to the right when compared with Fuzzy Interpolation Rule. Similarly, for Figure 11 (B), the output curve for Fuzzy Modifier interpolation Rule is shifted towards the left when a modifier is applied to the left valued Type-1 fuzzy set input. A very slight variation is observed in Figure 11 (C) when a modifier is applied to non-edge case valued Type-1 Fuzzy set input.

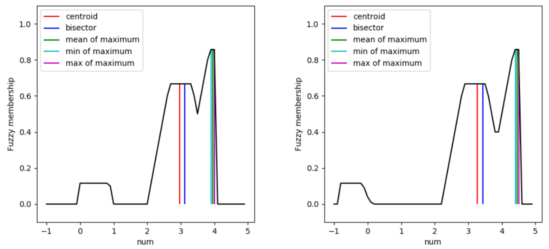


(a) Output Type-1 Fuzzy set after application of Fuzzy Interpolation Rule (b) Output Type-1 Fuzzy set after application of Fuzzy Modifier Interpolation Rule

Figure 9: Comparison between graphs obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Type-1 Fuzzy sets

Input Variable	**Value**
Tresbps	Extremely High
Chols	Rather High
Restecg	Relatively Terrible
Thalach	Relatively Terrible
Oldpeak	Relatively Terrible
Age	Extremely Old

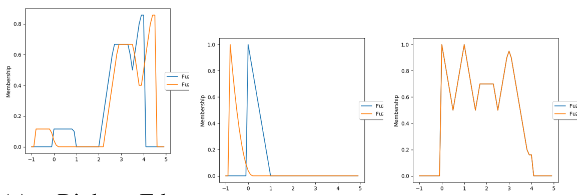
Table 2: Input Query



(a) A. Results obtained after defuzzification of the graphs obtained after application of Fuzzy IR

(b) B. Results obtained after defuzzification of the graphs obtained after application of Fuzzy Modifier IR

Figure 10: Comparison between Defuzzified results obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Type-1 Fuzzy sets

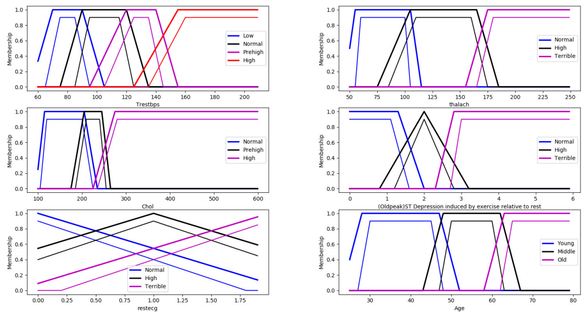


(a) Right Edge Case

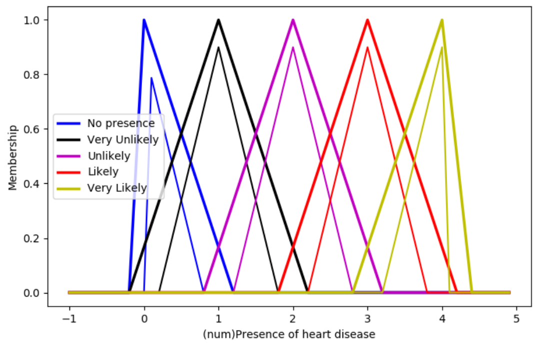
(b) Left Edge Case

(c) Non-Edge Case

Figure 11: Comparison between Fuzzy Interpolation rule (blue) and Fuzzy Modifier Interpolation Rule (Orange) for Type-1 Fuzzy sets



(a) Membership of Antecedents



(b) Membership of Consequent

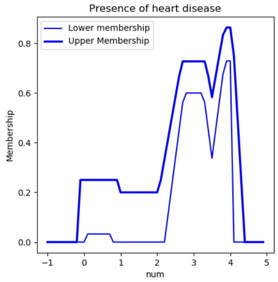
Figure 12: Type-2 Fuzzy Sets for different factors affecting Heart disease

4.2 Experiments and analysis for interval type-2 fuzzy sets

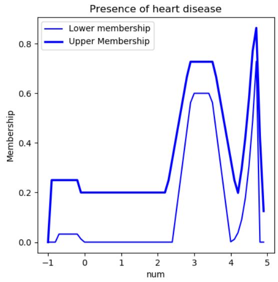
In the second half of the experiment, antecedents and consequent are represented using Interval Type-2 Fuzzy sets with triangular and trapezoidal memberships as shown in Figure 12. Similar to the experiments conducted on Type-1 Fuzzy sets, experiments on Interval Type-2 Fuzzy sets were conducted for three cases- right edge case, left edge case, and non-edge cases.

Similar to Type-1 Fuzzy sets, Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation Rule using min T-norm is applied to the query for right edge case as given in Table 2 and output curve obtained is depicted in Figure 13(A) and Figure 13(B) respectively for fuzzy interpolation rule and fuzzy modifier interpolation rule. We observe the graph obtained in Figure 13(B) is shifted to the left and right, taking care of the extreme left and right cases obtained on application of modifiers which is consistent with results obtained for Type-1 Fuzzy sets.

For type reduction of Interval Type-2 Fuzzy sets, the N-T algorithm was used, and for defuzzification, a similar methodology as discussed in the case of Type-1 fuzzy sets was used. Figure 14 shows the comparison between the results obtained for the Fuzzy Interpolation rule and Fuzzy Modifier Interpolation rule on defuzzification for the

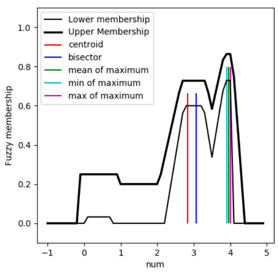


(a) Output Interval Type-2 Fuzzy set after application of Fuzzy Interpolation Rule

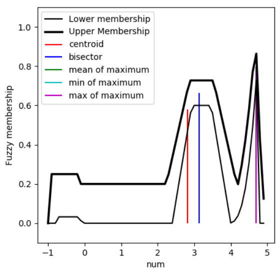


(b) Output Interval Type-2 Fuzzy set after application of Fuzzy Modifier Interpolation Rule

Figure 13: Comparison between graphs obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Interval Type-2 Fuzzy sets



(a) Output Interval Type-2 Fuzzy set after application of Fuzzy Interpolation Rule

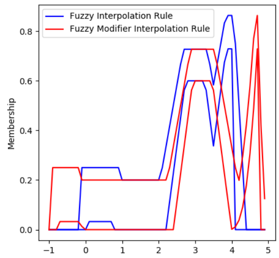


(b) Output Interval Type-2 Fuzzy set after application of Fuzzy Modifier Interpolation Rule

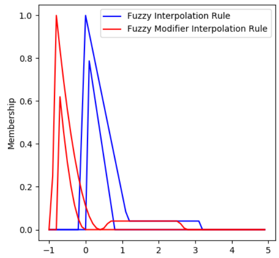
Figure 14: Comparison between defuzzified results obtained for Fuzzy Interpolation rule and Fuzzy Modifier Interpolation Rule for right edge case in Interval Type-2 Fuzzy sets

right edge case. In Figure 14 (B), the defuzzified values are shifted towards right for the mean of maximum, min of maximum, and max of maximum criteria. This corroborates the ability of our proposed interpolation rule to infer from edge cases efficiently.

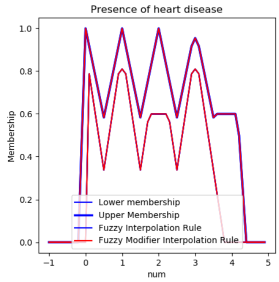
Similar to experiments performed on Type-1 Fuzzy sets, A comparison between Fuzzy Modifier Interpolation Rule and Fuzzy Interpolation Rule for all three cases- left edge case, non-edge case, and right edge case is shown for Interval Type-2 Fuzzy set. The blue curve in Figure 15 is obtained after the application of Fuzzy Interpolation Rule to Interval Type-2 Fuzzy sets, and Red curve is obtained after application of Fuzzy Modifier Interpolation Rule to Interval Type-2 Fuzzy sets. In Figure 15 (A), The modifier is applied to the right valued Interval Type-2 Fuzzy set input, and after application of Fuzzy Modifier Interpolation Rule the output curve is shifted to the right when compared with Fuzzy Interpolation Rule. Similarly, for Figure 15 (B),



(a) Right Edge Case



(b) Left Edge Case



(c) Non-Edge Case

Figure 15: Comparison between Fuzzy Interpolation rule (blue) and Fuzzy Modifier Interpolation Rule (red) for Interval Type-2 Fuzzy sets

the output curve for Fuzzy Modifier interpolation Rule is shifted towards the left when a modifier is applied to the left valued Interval Type-2 fuzzy set input. A very slight variation is observed in Figure 15 (C) when a modifier is applied to non-edge case valued Interval Type-2 Fuzzy set input.

5 Discussion

Table 3 shows the value of predicted attribute obtained after defuzzification of the output curve obtained as a result of the application of Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation Rule on both Type-1 Fuzzy set and Interval Type-2 Fuzzy set. No change in the values was obtained for the Fuzzy Interpolation Rule and Fuzzy Modifier Interpolation rule for the non edge case. Our proposed inference interpolation rule can effectively infer results from left edge case and right edge case, where the fuzzy sets are shifted towards extreme left and extreme right respectively on the application of modifiers, as shown in Table 3. Table 4 shows the final results obtained from the decision-making system after fuzzification of the results obtained from defuzzification.

The results obtained in table 4 depict the ability of our proposed inference rule to handle the left and right edge case effectively when modifiers are applied to Fuzzy set. Thus, Fuzzy Modifier Interpolation inference results in better modeling of variables used in a decision-making system and improved output curves and results on defuzzification.

	Right Edge Case	Left Edge Case	Non-Edge Case
Type-1 Fuzzy Set With Fuzzy Interpolation Rule	3.99999999999	-2.220446049250	1.808727539420
Type-1 Fuzzy Set with Fuzzy Modifier Interpolation Rule (Proposed Approach)	4.49999999999	-0.8	1.808727539420
IT-2 Fuzzy Set With Fuzzy Interpolation Rule	3.99999999999	0.0999999999999	1.92260915424
IT-2 Fuzzy Set With Fuzzy Modifier Interpolation Rule (Proposed Approach)	4.69999999999	-0.700000000000	1.92260915424

Table 3: Defuzzified Results

	Right Edge Case	Left Edge Case	Non-Edge Case
Type-1 Fuzzy Set With Fuzzy Interpolation Rule	Very Likely	Very Likely	Unlikely
Type-1 Fuzzy Set with Fuzzy Modifier Interpolation Rule (Proposed Approach)	Relatively Very Likely	Relatively Very Likely	Unlikely
IT-2 Fuzzy Set With Fuzzy Interpolation Rule	No Presence	No Presence	Unlikely
IT-2 Fuzzy Set With Fuzzy Modifier Interpolation Rule (Proposed Approach)	Rather No Presence	Rather No Presence	Unlikely

Table 4: Final Results Obtained

6 Advantage and future use

Fuzzy Interpolation Rule is quite a general rule that cannot deal with modifiers in Computing with words engine effectively. Until now, no efforts have been made to deal with modifiers separately in Type-1 Fuzzy sets and Interval Type-2 Fuzzy sets.

The Fuzzy Modifier Interpolation rule in Fuzzy Logic Inference engine provides the capability to introduce result as an extreme end fuzzy set that Fuzzy Interpolation rule cannot. $RELATIVELY\,HIGHLY\,EXCESS$, which is an extreme end fuzzy set that can be obtained on the application of this rule.

Given in Table 5 is the comparison between results drawn from the above experimentation on the UCLA dataset, which depict the improvement introduced by the application of Fuzzy Modifier Interpolation Rule.

Approach	Value
Fuzzy Modifier Interpolation Rule	Relatively Very Unlikely
Fuzzy Interpolation Rule	Very Unlikely

Table 5: Comparison Between Results Obtained

Fuzzy Modifier Interpolation Rule gives $RELATIVELY\,VERY\,UNLIKELY$ whereas Fuzzy Interpolation would have given only $VERY\,UNLIKELY$, This allows us to not only take care of edge cases in the modified fuzzy set but also get modified consequent as results. The use of modifiers on consequent gives the flexibility to express results beyond the pre-defined linguistic values for a fuzzy set.

Future use includes:

1. Incorporation of Fuzzy Modifier Interpolation Rule to Computing with Words inference engine to get better results.
2. Extending this rule to Fuzzy Weighted Average used in different decision making problem [24].
3. Incorporating this rule in fuzzy logic control that is used to regulate the movement of robots [25].
4. Applications in Medicine Diagnostic Systems when modifiers are to be applied [8].

7 Limitations

The Fuzzy Modifier Interpolation Rule produces precise and accurate results in most cases, but it fails when modifiers fail to satisfy the condition that the shift of Fuzzy sets on the application of modifier should be to its adjacent Fuzzy set only. This rule does not work for NOT modifier.

Proposed Fuzzy Modifier Interpolation rule for Type-1 and Interval Type-2 Fuzzy sets is applicable where only a single modifier is applied to the input given to Inference Engine. It fails when multiple modifiers are applied to the input. For example, X is $RELATIVELY\,SMALL$ is valid input; however, X is $RELATIVELY$ and $SOMEWHAT\,SMALL$ is invalid.

The Fuzzy Modifier Interpolation rule also fails when multiple connectives are used in the proposition like $X\,is\,RELATIVELY\,VERY\,SMALL$ or $Y\,is\,VERY\,VERY\,LARGE$. In the former proposition, modifier $RELATIVELY\,VERY$ is applied to the fuzzy set $SMALL$. In the latter proposition, the modifier $VERY\,VERY$ is applied to the fuzzy set $LARGE$. The shift in $SMALL$ and $LARGE$ curves is expected to extend beyond the adjacent fuzzy sets. As a result, the Fuzzy Modifier Interpolation Rule fails with multiple connectives. In future works, the Fuzzy Modifier Interpolation rule can be extended to include the application of multiple connectives as modifiers.

8 Conclusion

Computing with Words is gaining importance and becoming increasingly popular. It has many applications in science, and varied industries [29][30][31]. The inference engine or different techniques of inferencing results like the Fuzzy Interpolation rule in CW play a significant role in producing results. However, it has limitations when applied to modified Fuzzy sets. Therefore, if some modifications to the inference rules can give better outputs when dealing with modified Fuzzy sets, then they have a significant impact on all the fields where CW is being implemented. On a concluding note, the proposed Fuzzy Modifier Interpolation rule for Type-1 Fuzzy Sets and Interval Type-2 is an important milestone because of its ability to take care of left and right extreme cases when dealing with modified fuzzy sets. This results in improvement of the output curves obtained and better results on defuzzification. Its inclusion in the Inference rules would be beneficial for every user who applies the CW model.

Acknowledgement

We want to extend our sincere thanks to Aashi Jain, Divya Gupta, and Garima Gupta for their inputs during idea conception and our anonymous reviewers for their invaluable feedback.

References

[1] L.A. Zadeh , Fuzzy logic= computing with words, IEEE transactions on fuzzy systems 4.2, 103-11(1996)

[2] L.A. Zadeh, Key roles of information granulation and fuzzy logic in human reasoning, Concept formulation and computing with words.Proceedings of the Fifth IEEE International Conference , Vol. 1.(1996)

[3] J. Mendel, J. Lawry,L.A. Zadeh. Foreword to the special section on computing with words, IEEE Transactions on Fuzzy Systems, 18.3,437-440 (2010)

[4] L. A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning—I, Inform. Sci., 8,pp. 199–249 (1975)

[5] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning—II, Information sciences, 8.4, 301-357 (1975)

[6] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning-III, Information sciences, 9.1, 43-80 (1975)

[7] J.M. Mendel, Computing with words, when words can mean different things to different people, ICSC Symposium on Fuzzy Logic and Applications, pp. 158–164 (June 1999)

[8] C. Schuh, Fuzzy set and their application in medicine, NAFIPS Annual Meeting of the North American Fuzzy Information Processing Society (2005)

[9] L.A. Zadeh, A new direction in AI: Toward a computational theory of perceptions, AI magazine, 22.1, 73 (2001)

[10] L.A. Zadeh, Fuzzy sets as a basis for a theory of possibility, Fuzzy sets and systems, 1.1, 3-28 (1978)

[11] L.A. Zadeh, Fuzzy logic, Computer, 21.4, 83-93 (1988)

[12] Zadeh, Lotfi A. "A fuzzy-set-theoretic interpretation of linguistic hedges." (1972): 4-34.

[13] Hellendoorn, Johannes. "Reasoning with fuzzy logic." (1990).

[14] De Cock, Martine, and Etienne E. Kerre. "Fuzzy modifiers based on fuzzy relations." Information Sciences 160.1-4 (2004): 173-199.

[15] De Cock, Martine, and Etienne E. Kerre. "A context-based approach to linguistic hedges." International Journal of Applied Mathematics and Computer Science 12 (2002): 371-382.

[16] B. Bouchon-Meunier and C. Marsala, "Linguistic modifiers and measures of similarity or resemblance," Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569), 2001, pp. 2195-2199 vol.4, doi: 10.1109/NAFIPS.2001.944410.

[17] Bin-Da Liu, Chuen-Yau Chen and Ju-Ying Tsao, "Design of adaptive fuzzy logic controller based on linguistic-hedge concepts and genetic algorithms," in IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 31, no. 1, pp. 32-53, Feb 2001, doi: 10.1109/3477.907563.

[18] Ying, Mingsheng, and Bernadette Bouchon-Meunier. "Approximate reasoning with linguistic modifiers." International journal of intelligent systems 13.5 (1998): 403-418.

[19] Bernadette Bouchon-Meunier. 1992. Fuzzy logic and knowledge representation using linguistic modifiers. Fuzzy logic for the management of uncertainty. John Wiley & Sons, Inc., USA, 399–414.

[20] Cordón, Oscar, María José del Jesus, and Francisco Herrera. "Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods." International Journal of Intelligent Systems 13.10-11 (1998): 1025-1053.

[21] S. Medasani and R. Krishnapuram, "A fuzzy approach to content-based image retrieval," FUZZ-IEEE'99. 1999 IEEE International Fuzzy Systems. Conference Proceedings (Cat. No.99CH36315), 1999, pp. 1251-1260 vol.3, doi: 10.1109/FUZZY.1999.790081.

[22] H. Nakajima, T. Sogoh and M. Arao, "Fuzzy database language and library-fuzzy extension to SQL," [Proceedings 1993] Second IEEE International Conference on Fuzzy Systems, 1993, pp. 477-482 vol.1, doi: 10.1109/FUZZY.1993.327514.

[23] P. Bosc and O. Pivert. 1994. Fuzzy queries and relational databases. In Proceedings of the 1994 ACM symposium on Applied computing (SAC '94). Association for Computing Machinery, New York, NY, USA, 170–174. DOI:https://doi.org/10.1145/326619.326690

[24] F. Liu , J. Mendel, Aggregation using The Fuzzy Weighted Average as computed using Kernel-Mendel Algorithm. IEEE transactions on fuzzy systems, vol. 16 (february 2008)

[25] Aguirre, Eugenio, and Antonio González, Fuzzy behaviors for mobile robot navigation: design, coordination and fusion, International Journal of Approximate Reasoning, 25.3 ,255-289 (2000)

[26] L.A. Zadeh, Fuzzy Sets, information and control, 8, 338–353 (1965)

[27] David Aha, Heart Disease Data Set

[28] Y. Yuan, M.J. Shaw, Induction of fuzzy decision trees,Fuzzy Sets and Systems, Vol 6, Pages 125-139 (1995)

[29] L.A. Zadeh, Some reflections on information granulation and its centrility in granular computing, computing with words, the computational theory of perceptions and precisiated natural language, Studies in Fuzziness and Soft Computing,95

[30] L.A. Zadeh, A summary and update of "fuzzy logic", Granular Computing (GrC), 2010 IEEE International Conference

[31] A. Jain, Applications of computing with words in medicine: Promises and potential, Fuzzy Systems (FUZZ-IEEE), 2017 IEEE International Conference