All Modern AI Needs Is Intuitionistic Fuzzy

Vladik Kreinovich[1], Olga Kosheleva[2], Victor Timchenko[3], and Yuriy Kondratenko[4]

[1,2]Departments of [1]Computer Science and [2]Teacher Education
University of Texas at El Paso
500 W. University, El Paso, Texas 79968, USA
olgak@utep.edu, vladik@utep.edu
[3]Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine
vl.timchenko58@gmail.com
[4]Petro Mohyla Black Sea National University, Mykolaiv, Ukraine
y_kondrat2002@yahoo.com

1. Modern AI: Successes and Challenges

- In the last decades, AI techniques have reached spectacular successes.
- Large Language Models like GPT have become part of our everyday life:
  - they help edit papers,
  - they help plan classes,
  - they help design new problems for student exams,
  - they help answer difficult questions – be it a medical question or a question related to research.
- But there is a serious problem with the AI answers: a large percentage of these answers are wrong.
- In modern AI, such wrong answers are known as *hallucinations*.
- When ChatGPT first became public, about 15% of its answers were wrong.
- Now this proportion has been brought down to 5%.

2. Modern AI: Successes and Challenges (cont-d)

- This is much better, but for many situations, this is still unacceptable.
- A student may successfully pass a long test with 20 question if he/she answers one of them (5%) wrong.
- But in emergency medical situations, 5% means one of the 20 patients may die because of the wrong recommendation.
- In space exploration, this means one out of 20 launches of piloted spaceships may end in a disaster.
- This is clearly unacceptable.
- So, what can we do?

3. What Causes AI Hallucinations and How to Avoid Them: A Brief Reminder

- To have examples of hallucination, there is no need to limit ourselves to complex questions.
- This phenomenon can be observed already when we ask simple binary "yes"-"no" questions.
- To each such question, the AI systems replies "yes" or "no".
- These answers are supplemented with the system's degree of confidence in its answer.
- Usually, these degrees are obtained by using the so-called softmax algorithm.
- For two possible outcomes, this algorithm produces two numbers that add up to 1.

4. What Causes AI Hallucinations and How to Avoid Them: A Brief Reminder (cont-d)

- Specifically, if the neural network produced:
  - the degree x_1 corresponding to the "yes" answer and the degree x_0 corresponding to the "no" answer,
  - then the softmax algorithm assigns the following degrees of confidence d_i to these answers:

$$d_1 = \frac{\exp(\alpha \cdot x_1)}{\exp(\alpha \cdot x_1) + \exp(\alpha \cdot x_0)} \text{ and } d_0 = \frac{\exp(\alpha \cdot x_0)}{\exp(\alpha \cdot x_1) + \exp(\alpha \cdot x_0)}.$$

- Here, $\alpha > 0$ is some parameter.
- Similarly:
  - for classification problems in which we need to assign an object to one of several n classes,
  - the system returns the degree of confidence for each class.
- These degrees that add up to 1.

5. What Causes AI Hallucinations and How to Avoid Them: A Brief Reminder (cont-d)

- Specifically:
  - if the neural network returns the value x_i for the i-th class,
  - then the resulting degree of confidence d_i is computed as follows:

$$d_i = \frac{\exp(\alpha \cdot x_i)}{\exp(\alpha \cdot d_1) + \ldots + \exp(\alpha \cdot x_n)}.$$

- Let us get back to the binary case.
- According to the above formulas:
  - when the AI system has an overwhelming number of arguments for a statement and very few arguments against,
  - it replies "yes" with a high degree of confidence.
- So far so good, this is exactly what we would expect.
- But what if we only have one weak argument in favor and no arguments against (or an argument against that is even weaker).

6. What Causes AI Hallucinations and How to Avoid Them: A Brief Reminder (cont-d)

- In this case, a human expert would be hesitant to conclude "yes" or "no".
- This expert would most probably say:
  - that he/she does not have enough information about the situation, and
  - that a further study is needed.
- But in such a situation, an AI tool bravely concludes "yes".
- Why? Because most such tools do not even have a choice of giving the "don't know" answer.
- What we therefore need is to enable the AI tools to have,
  - for each binary question,
  - a third possible answer in addition to its "yes" and "no" answers: the answer "don't know".

7. This Is Exactly What Intuitionistic Fuzzy Technique Is About

- A similar situation happened in the past.
- That situation motivated the design of intuitionistic fuzzy techniques.
- Namely, in the original fuzzy logic, we only had one number to describe our degree of confidence s in a statement S.
- In this logic, the degree of confidence in the negation $\neg S$ of a statement S:
  - is simply computed as $1 - s$,
  - i.e., is computed as one minus the degree of confidence in a statement itself.

8. This Is Exactly What Intuitionistic Fuzzy Technique Is About (cont-d)

- As a result, if we have many arguments in favor of a statement, twice more than arguments against:
  - then we take $2/3$ as our degree of confidence in a statement, and,
  - correspondingly, $1 - 2/3 = 1/3$ as our degree of confidence in its negation.
- This makes sense.
- However:
  - if we simply have two very weak arguments in favor of the statement and one equally weak argument against,
  - then, in the traditional fuzzy formalism, we should also assign the same degrees $2/3$ and $1/3$.
- This does not make much sense here.

9. This Is Exactly What Intuitionistic Fuzzy Technique Is About (cont-d)

- To avoid such a counterintuitive situation, intuionistic fuzzy logic provides *three* degrees that add up to 1.
- The additional degree describes to what extent we do not know the correct answer.

10. So, What We Need Is to Incorporate Intuitionistic Fuzzy Logic into AI Tools

- At present, with respect to degrees of confidence, the current AI tools operate on the level of the traditional fuzzy logic.
- What we need is to move them to the next level – corresponding to intuitionistic fuzzy logic.

11. How Can We Do It: Ideal Situation

- Why do AI tools say “yes” or “no” instead of “don’t know”?
- There is no malicious intent, no conspiracy behind this, this is just how these tools are trained.
- During the training, the main criterion is, in effect, the proportion of correct answers.
- So suppose that the AI tools encounters several (n) situations in which the probability of “yes” answer is slightly larger than half.
- In this case, from the viewpoint of the objective function, it is beneficial for this tool to reply “yes” to all of them.
- Indeed, this will increase the average number of correct answers by at least $n/2$.
- To avoid such situations:
  - instead of number of correct answers,
  - we should maximize the overall utility of all the answers.

12. How Can We Do It: Ideal Situation (cont-d)

- In situations with high uncertainty – e.g., in medical situations – both "yes" and "no" answer may be risky.
- In such cases, it is better to perform some additional tests before starting treatment.
- At present, on each training example, the only information that the AI tool gets is the correct answer – "yes" or "no".
- What we need is to supplement this information with the expected utility of giving one of the three answers:

  "yes", "no", and "don't know".

- For example:
  - if the expert is almost 100% confident about a diagnosis,
  - then the "don't know" answer would mean wasting time and resources on additional tests.

13. How Can We Do It: Ideal Situation (cont-d)

- On the other hand:
  - when the expert is not very confident,
  - then tests may be justified, while both "yes" and "no" answers may be dangerous.

14. Comments

- Another example of when an imperfect choice of the objective function worsens the AI tool's performance is described in a recent paper:
  - if we use user satisfaction as a performance criterion,
  - then the tool becomes less accurate,
  - it often provides a wrong answer – that pleases the user – instead of the correct answer.
- It is also worth mentioning that:
  - the need to make the objective function more adequate to improve the situation
  - is one of the main ideas behind Buddhism.

15. Comments (cont-d)

- Indeed, according to Buddhism, a significant part of our suffering comes from having:
  - unrealistic, unnecessary, inadequate desires,
  - i.e., in mathematical terms, unrealistic, unnecessary, inadequate objective functions.
- For example:
  - instead of enjoying simple healthy pleasures of life,
  - people suffer because they want to live as well as (or even better than) their richest neighbors.

16. While We Are Waiting for the Ideal Situation, What Can We Do Now?

- It would nice to have such extended data and to train a neural network on such an extended data.
- However, at present, we do not have such data.
- What can we do now, while waiting for the ideal situation to occur?
- One possibility is to us a modification of softmax proposed (and justified) in a recent paper.
- For classification into n classes:
  - instead of the usual softmax formula,
  - we should use the modified formula

$$d_i = \frac{\exp(\alpha \cdot x_i)}{\exp(\alpha \cdot x_1) + \ldots + \exp(\alpha \cdot x_n) + C}.$$

- Here, $C > 0$ is an additional constant.

17. While We Are Waiting for the Ideal Situation, What Can We Do Now (cont-d)

- In this case:
  - in addition to n probabilities of belonging to the corresponding class,
  - we also have the additional probability
  $$p_0 = \frac{C}{\exp(\alpha \cdot x_1) + \ldots + \exp(\alpha \cdot x_n) + C};$$
  - it is the probability that "don't know" answer is appropriate.
- In such situation:
  - instead of the original equality $p_1 + \ldots + p_n = 1$,
  - we have a modified equality $p_1 + \ldots + p_n + p_0 = 1$.

18. While We Are Waiting for the Ideal Situation, What Can We Do Now (cont-d)

- In particular, for a binary question:
  - in addition to the degrees of confidence d_1 and d_0 in, correspondingly, "yes" and "no" answers:

$$d_1 = \frac{\exp(\alpha \cdot x_1)}{\exp(\alpha \cdot x_1) + \exp(\alpha \cdot x_0) + C} \text{ and } d_0 = \frac{\exp(\alpha \cdot x_0)}{\exp(\alpha \cdot x_1) + \exp(\alpha \cdot x_0) + C},$$

  - we also have the following degree of confidence in the "don't know" answer:

$$d_a = \frac{C}{\exp(\alpha \cdot x_1) + \exp(\alpha \cdot x_0) + C}.$$

- These three numbers that add up to 1 is what the AI tools should generate.
- This is very easy to implement – all we have to do is to make a minor modification to the last – softmax – layer.

19. General Comment: Two Values Good, Three Values Better

- In effect, what we are proposing is going:
  - from the current techniques – which is, in effect, a fuzzification of the usual 2-valued logic,
  - to a modified intuitionistic-fuzzy-type techniques that fuzzifies a three-valued logic, with values "yes", "no", and "unknown".
- Let us list additional advantages of using 3-values logic.
- From the *mathematical* viewpoint, 3-valued logic more adequately describes the truth values of mathematical statements.
- Indeed, some of them are true, some are false, and some – according to the famous Goedel's theorem – are undecidable.
- From the *computational* viewpoint: according to Knuth, 3-valued number systems provide the most effective computations.
- Specifically, the most effective computations correspond to a system that uses digits 1, -1, and 0.

20. General Comment: Two Values Good, Three Values Better (cont-d)

- These digits directly correspond to “true”, “false”, and “unknown”.
- Another computational advantage of the 3-valued interpretation is that it is possible:
  - to use combinations of 3 basic colors – red, green, and blue,
  - to effectively perform analog computations, in particular, fuzzy-related analog computations.
- From the *psychological* viewpoint:
  - in many cases, when a decision maker is deciding between two solutions each of which has advantages and limitations,
  - there is actually yet another (third) alternative that has more advantages and fewer limitations.
- Looking for such an alternative is often a good decision strategy.

21. General Comment: Two Values Good, Three Values Better (cont-d)

- From the *dynamical* viewpoint: a 3-part system is more stable.
- This is true for 3 competing superpowers or 3 competing species, etc.
- When one of the parties become too powerful, the other two team against it.
- On the other hand, similar 2-part systems are unstable:
  - if one of the parts becomes stronger,
  - it can extinguish the other one.

22. 3-Valued (Intuitionistic-Fuzzy) Approach May Be More Adequate, But It Is More Computationally Complex

- Our stop-gap solution is easy to implement.
- However, a more adequate implementation of the three-valued approach requires a drastic increase in computational complexity.
- Let us first give a natural example of such an increase.
- This example relates to the fact that we rarely make a decision based on the answer to a single question.
- We usually ask several questions.
- When making decision, we take into account the answers to all these questions – and degrees of confidence in these answers.
- For example, we can diagnose a certain disease if we observe two symptoms S_1 and S_2.

23. 3-Valued Approach Is More Computationally Complex (cont-d)

- Suppose that we are not 100% confident about each of these symptoms.
- Suppose that our degrees of confidence in these symptoms are s_1 and s_2.
- Then our degree of confidence that the patient has the given disease is equal to $f_\&(s_1, s_2)$,
- Here, $f_\&(a, b)$ is an appropriate "and"-operations – also known as t-norm.
- For example, we can have $f_\&(a, b) = a \cdot b$.
- In this case our degree of confidence is $s_1 \cdot s_2$.
- The same diagnosis may be also justified if we have two other symptoms S_3 and S_4 at the same time.

24. 3-Valued Approach Is More Computationally Complex (cont-d)

- In this case, our degree of confidence that a patient has this disease is equal to our degree of confidence in a more complex formula

$$(S_1 \,\&\, S_2) \vee (S_1 \,\&\, S_4).$$

- We may have even more complex formulas.
- We can compute the degree of confidence in each such formula if we use:
  - "and"-operation instead of the simple "and',
  - "or"-operation $f_\vee(a, b)$ (also known as t-conorm) instead of the simple "or", and
  - negation operation $f_\neg(a)$ instead of the simple negation.
- For example, for the above 4-symptom formula, the corresponding degree is equal to $f_\vee(f_\&(s_1, s_2), f_\&(s_3, s_4))$.
- In the traditional fuzzy logic, the resulting degree is easy to compute.

25. 3-Valued Approach Is More Computationally Complex (cont-d)

- However, in the intuitionistic fuzzy logic, this computation problem becomes NP-hard.
- This means that:
  - unless P = NP – which most computer scientists believe to be not possible,
  - no feasible algorithm is possible that would solve all instances of this problem.
- Indeed, a particular possibility in the intuitionistic fuzzy case is:
  - when we have no information at all about each statement,
  - i.e., when both positive and negative degrees of certainty are zeros.
- In this case:
  - if the logical formula F is always false,
  - then the resulting degree of confidence in its being false is 1.

26. 3-Valued Approach Is More Computationally Complex (cont-d)

- However:
  - if the formula is true for some values of its propositional variables,
  - it may be true,
  - so our degree of confidence that it is false is smaller than 1.
- And the problem of checking whether a propositional formula is always false is known to be NP-hard,
- So the problem of exactly computing the resulting degrees is also NP-hard.
- There are other cases when going from 2 alternatives to 3 drastically increases the computational complexity.

27. 3-Valued Approach Is More Computationally Complex (cont-d)

- For example, a simple feasible algorithm can check whether:
  - a map (or, more generally, a graph) can be colored in 2 colors
  - so that neighboring countries (or neighboring vertices) have different colors.
- However, the problem of coloring a graph in 3 colors is already NP-hard.
- Interestingly, the standard proof of its NP-hardness:
  - uses "true", "false", and "unknown" as the names of the three colors, and
  - uses the corresponding logical intuition in its proof.
- This increase in complexity is well-known.

28. 3-Valued Approach Is More Computationally Complex (cont-d)

- Deleting such unknown examples at the first stage of learning is a known pedagogical idea:
  - at first, we teach the kids a simplified black-and-white description of the state of the world, what is good and what is bad, and
  - only later teach them a more realistic and more nuanced world picture.
- Not surprisingly, a similar tactic seems to work when we teach a neural network:
  - first, we only train it on clear examples, and
  - only later, we include more uncertain examples in our training.
- There are other examples of how going from 2 to 3 makes computations more complex.

29. 3-Valued Approach Is More Computationally Complex (cont-d)

- For example, in celestial mechanics:
  - a 2-body problem has an explicit solution, while
  - a 3-body problem is difficult to solve.
- Similarly, 2-person dynamics is easy to analyze and easy to optimize.
- Namely, moderately good attitudes towards each other make everyone feels good.
- On the other hand, for a 3-person interactions, similar attitude often lead to negative feelings.

30. Acknowledgments

This work was supported by:

- the AT&T Fellowship in Information Technology,
- the Institute for Risk and Reliability, Leibniz Universitaet Hannover, Germany,
- the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) Focus Program SPP 100+ 2388, Grant Nr. 501624329,
- the European Union under the project ROBOPROX (No. CZ.02.01.01/00/22 008/0004590),
- the Center of Excellence in Econometrics, Faculty of Economics, Chiang Mai University, Thailand,
- the Ho Chi Minh City University of Banking, Vietnam, and
- Thang Long University, Hanoi, Vietnam.