Fuzzy Logic Acquisition with Multi-modal Contexts

Maxwell J. Jacobson
Email: jacobs57@purdue.edu

Yexiang Xue
Email: yexiang@purdue.edu

1 Introduction

The integration of constraint programming (CP) and machine learning (ML) requires a formal language capable of bridging abstract, constraint-based reasoning with the fuzzy, multi-modal complexities of real-world environments. Formal logics have proven effective in structured domains. However, applying them to real-world data often leads to limitations. For instance, specifying the precise desired location of a microwave in a kitchen using hard constraints is impractical, as spatial relations with contextual nuances (convention, aesthetics, etc) cannot feasibly be expressed.

Fuzzy and probabilistic logic offers a promising alternative, utilizing degrees of membership rather than rigid classifications [8, 9, 10]. Additionally, research has explored integrating fuzzy logic with ML and CP, such as using classification systems for constraint acquisition [7, 3]. However, current fuzzy logic systems still usually depend on human interpretation, making them difficult to automate effectively in multi-modal applications, among others.

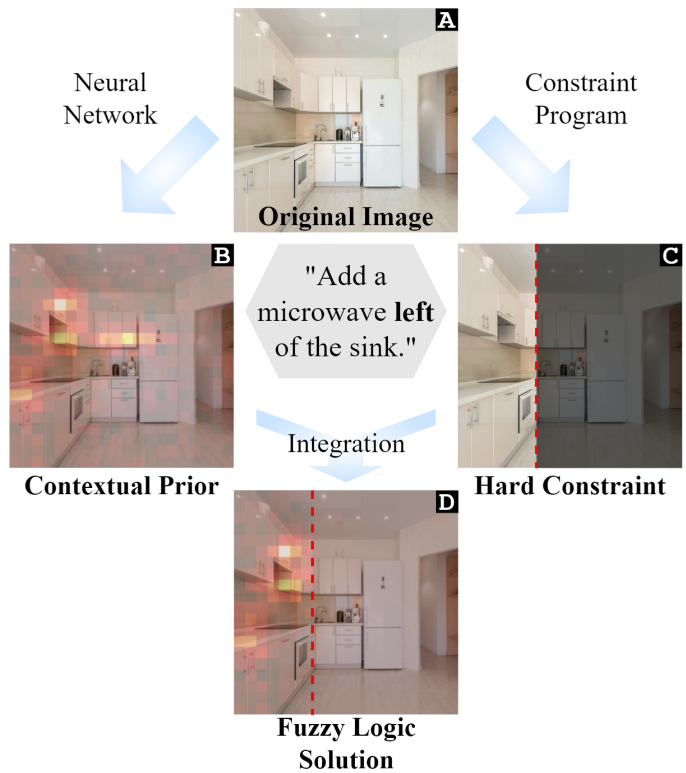

Figure 1: Fuzzy constraint acquisition for placing a microwave in a scene image based on the text "left of the sink." **(A)** The original image. **(B)** A heatmap showing the soft contextual component: a neural network infers a distribution over microwave locations based on the image and text context. **(C)** The hard component: the area to the right of the red line is excluded by the "left of" condition. **(D)** The combined contextual fuzzy logic solution.

We propose a novel fuzzy logic framework designed to operate seamlessly in multi-modal contexts. By integrating CP and neural networks, this approach aims to enable automated reasoning in complex environments while maintaining the flexibility of fuzzy representations. This will be done by casting **predicates as hybrid entities**, where part of their meaning is derived from hardcoded binary values, and part from context-informed neural networks outputting probability distributions. In this position paper, we propose this logic on an exemplar spatial reasoning domain, and then discuss some methods of implementation.

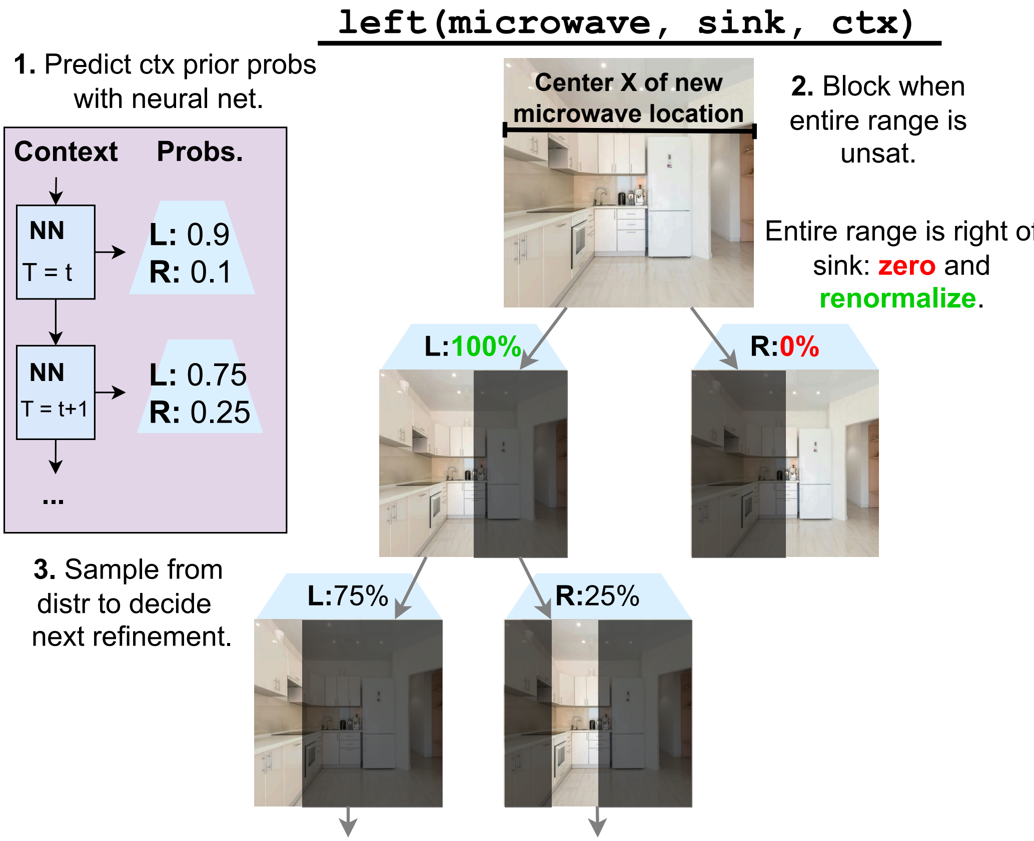

Figure 2: The *iterative refinement* method for implementing our contextual fuzzy logic. The 3 steps detail probability inference from multi-modal context, blocking and pruning constrained probabilities, and sampling to make a refinement step.

2 Contextual Fuzzy Logic Under Multi-modal Data

Focusing on spatial logic as an example case, we develop this contextual fuzzy logic for the task of placing textually-defined objects into a scene image. This task is a good example case because it requires interaction with both natural language and visual data to determine appropriate spatial locations for objects, such as placing a microwave in a kitchen already containing various furnishings (See Figure 1). It is fundamentally spatial, relying on constraints to enforce relationships between objects as perceived from the camera's perspective. While spatial reasoning is just one possible application of contextual fuzzy logic, it exemplifies how we can integrate multi-modal data to model real-world complexities.

Objects and Spatial Components. Objects represent physical entities, denoted as o_1, o_2, o_3, etc., with spatial components x and y defining their centers relative to the camera. For example, o_2^x represents the x-coordinate of object o_2. Positive directions are defined as rightward for x and downward for y. These components serve as the foundation for reasoning about spatial relationships.

Predicates and Multi-modal Context. Spatial predicates such as $leftof$, $rightof$, $above$, and $below$ define relationships between objects within a given multi-modal context ctx. Instead of resolving to a binary result, these predicates yield a probability distribution over an object's coordinates. For example, $leftof(o_1, o_2, ctx)$ returns a probability distribution over x-values, where $P(o_1^x \geq o_2^x) = 0$, and the remaining probabilities depend on ctx. Formally, given a potential assignment k:

$$leftof(o_1, o_2, ctx) \rightarrow P(o_1^x = k \mid o_1^x < o_2^x, ctx)$$

Basic Operands. We include three primary operands for combining predicates:

- $A \wedge B = norm(P(A) \times P(B))$ – Conjunction ($\wedge$): probabilities are multiplied and normalized.
- $A \vee B = norm(max(P(A), P(B)))$ – Disjunction ($\vee$): maximum probabilities are selected and normalized.
- $\neg A = 1 - P(A)$ – Negation ($\neg$): probabilities are inverted.

3 Learning the Contextual Fuzzy Logic

One effective way to implement this combined approach is through a process of *iterative refinement* [6]. In this method, a neural network refines a range of possible values for each coordinate through steps of halving the range and sampling one half to continue down, in a variant of SampleSearch [5]. During each step, the hard components are enforced by "zeroing-out" branches of the search tree that violate the constraints, followed by re-normalizing the remaining branches. Dynamic enforcement of constraints allows the system to prune invalid solutions early in the search process, reducing computational overhead while maintaining interpretability in the intermediate steps. This approach (diagrammed in Figure 2) allows constraints to be enforced dynamically throughout the search process, avoiding invalid solutions early and reducing computational overhead. Interpretability in the intermediate steps is also maintained.

While iterative refinement offers significant advantages, it is not the only method within this family. Another might involve embedding a constraint optimization layer directly within a neural network using differentiable convex optimization solvers [4, 2, 1]. This single-step approach allows for optimization to occur directly during network training. However, this method limits constraints to convex ones, which may restrict the types of relationships that can be modeled and may also offer a less tightly integrated approach compared to iterative refinement.

Submission Details

Submitted to: Position Papers Track.
Institution: Purdue University.
Attending: Both authors will be attending.

References

[1] Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable convex optimization layers. In *Advances in Neural Information Processing Systems*, pages 9558–9570, 2019.

[2] Akshay Agrawal, Robin Verschueren, Steven Diamond, and Stephen Boyd. A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1):42–60, 2018.

[3] Christian Bessiere, Remi Coletta, Eugene C. Freuder, and Barry O'Sullivan. Leveraging the learning power of examples in automated constraint acquisition. In Mark Wallace, editor, *Principles and Practice of Constraint Programming – CP 2004*, pages 123–137, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[4] Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

[5] Vibhav Gogate and Rina Dechter. Samplesearch: Importance sampling in presence of determinism. *Artificial Intelligence*, 175(2):694–729, 2011.

[6] Maxwell J. Jacobson and Yexiang Xue. Integrating symbolic reasoning into neural generative models for design generation. *Artificial Intelligence (AIJ)*, 2025. To appear.

[7] S. D. Prestwich, E. C. Freuder, B. O'Sullivan, and D. Browne. Classifier-based constraint acquisition. *Annals of Mathematics and Artificial Intelligence*, 89(7):655–674, 2021.

[8] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine learning*, 62:107–136, 2006.

[9] Lotfi A. Zadeh. Fuzzy logic. In T.-Y. Lin, W. Pedrycz, and A. Bargiela, editors, *Granular, Fuzzy, and Soft Computing*, pages 19–49. Springer-Verlag, 2009. Originally published in Meyers, R. A. (Ed.), *Encyclopedia of Complexity and Systems Science*.

[10] Yuxuan Zhang, Xiangzhi Bai, Ruirui Fan, and Zihan Wang. Deviation-sparse fuzzy c-means with neighbor information constraint. *IEEE Transactions on Fuzzy Systems*, 27(1):185–199, 2019.