Concept Learning Based on Human Interaction and Explainable AI

Blake Ruprecht[a,b], Derek T. Anderson[a,b], Fred Petry[c], James Keller[a,b], Christopher Michael[c], Andrew Buck[a,b] , Grant Scott[a,b], and Curt Davis[a,b]

[a]Electrical Engineering and Computer Science Dept, University of Missouri, Columbia, MO
[b]Center for Geospatial Intelligence, University of Missouri, Columbia, MO
[c]U.S. Naval Research Laboratory, Stennis Space Center, Stennis, MS

ABSTRACT

In this article, we explore the role and usefulness of parts-based spatial concept learning about complex scenes. Specifically, we consider the process of teaching a spatially attributed graph how to utilize parts-detectors and relative positions as attributes in order to learn concepts and to produce human oriented explanations. First, we endow the graph with parts detectors and relative positions to determine the possible range of attributes that will limit the types of concepts that are learned. Next, we enable the graph to learn concepts in the context of recognizing structured objects in imagery and the spatial relations between these objects. As the graph is learning concepts, we allow human operators to give feedback on attribute knowledge, creating a system that can augment expert knowledge for any similar task. Effectively, we show how to perform online concept learning of a spatially attributed graph. This route was chosen due to the vast representational capabilities of attributed graphs, and the low-data requirement of online learning. Finally, we explore how well this method lends itself to human augmentation, leveraging human expertise to perform otherwise difficult tasks for a machine. Our experiments shed light on the usefulness of spatially attributed graphs utilizing online concept learning, and shows the way forward for more explainable image reasoning machines.

Keywords: concept learning, graph learning, spatial relations, human interaction, explainable ai

Acknowledgement: Petry and Michael would like to thank the Naval Research Laboratory's Base Program for sponsoring this research.

1. INTRODUCTION

In the last decade, we have seen an exponential rise in research and funding of machines that can perform visual object detection exceptionally well. However, this advancement, primarily in regards to standard measures of performance like F1, has originated from techniques that function as correlation machines, rather than concept learners. While a success no doubt, these so-called black box machines arguably generate more problems than solutions. One major problem is the explainability and accountability of these machines, which generate no natural explanations and don't promise similar outcomes under similar circumstances. Explainable AI (XAI) attempts to rectify this in one way by developing novel techniques to perform concept learning rather than correlation learning. In essence, concept learning is the search for attributes that are used to distinguish exemplars from non-exemplars of a certain category. This is inherently explainable due to the ability to point to the attributes that did or did not lead to a decision made by the machine. Furthermore, the inclusion of a human in the loop provides oversight and faster machine learning from the interaction of the human and machine. While broad in scope, there are many application areas that can benefit from concept learning based on human interaction and XAI, including computer vision, geospatial sensing, pattern recognition, object detection, natural language processing, and more. In the context of this paper, we are less focused on specific application, and more focused on the dialogue that is created between human and machine when using explainable AI techniques. An explainable AI machine can naturally generate explanations for decisions, a human in the loop can evaluate these explanations and decisions and provide feedback, and the machine accepts the feedback to improve concept definitions, leading to better explanations and performance.

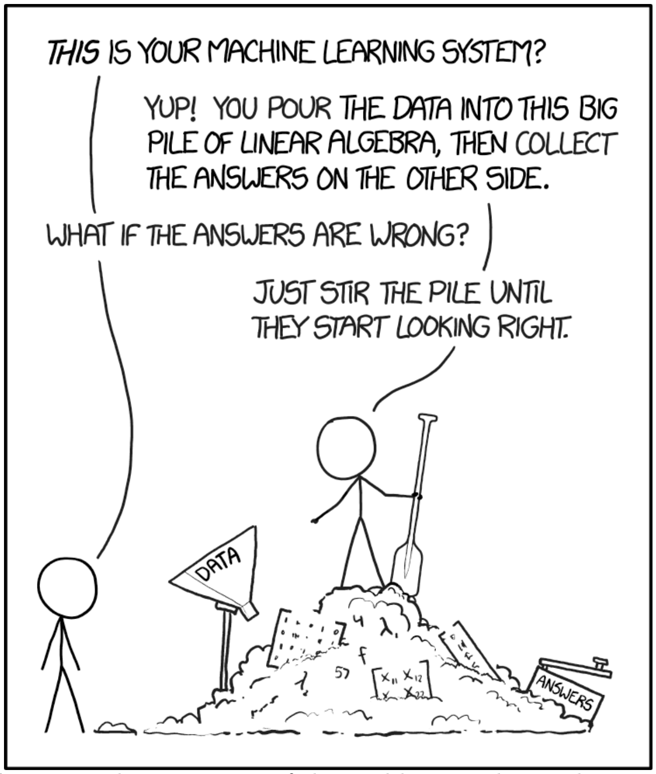


Figure 1: A humorous XKCD comic showing some of the problems with correlation machine learning techniques and their inherent lack of explainability. From xkcd.com/1838/.

The contributions of this article are as follows. First, we discuss the high-level benefits and drawbacks of concept learning based on human interaction and explainable AI, along with discussing some potential application areas. Second, we describe the low-level methods and techniques of our machine, and the spatially attributed graph that combines these techniques into one learner. Third, we illustrate the explanation-feedback loop of concept learning with the human in the loop, highlighting the knowledge that the machine is using, the generated explanations, how humans can provide feedback, and the improvements for both human and machine that this technique provides. Finally, we provide an example of what concept learning of a spatially attributed graph with human interaction would look like on a synthetic concept example, and show how this method would benefit a scenario where a difficult false positive is encountered.

The remainder of this article is organized as follows. In Section 2, we discuss the high level overview of the method, and some specific applications. Section 3 discusses concept learning, spatially attributed graphs, human interaction, parts detectors, and spatial relations. Section 4 discusses the overall method and how the specific methods are combined. Section 5 provides an example of the method, followed by a summary and future work.

2. BIG PICTURE

Computational solutions to problems do not always make intuitive sense to humans, especially when the pile of equations used to solve the problem gets larger and larger (see Figure 1). Machines solve tasks using the mathematical techniques we endow them with. When the techniques aren't interpretable, there is little hope for creating an explanation. Fuzzy uncertainty is an effective way of dealing with many of the problems that come up in machine learning. Unfortunately, many methods cannot adequately handle processes with fuzzy uncertainty, which leads to worse decisions and explanations. However, when interpretable math techniques are used, explanations naturally exist, and because of this a human can interpret what the machine is doing, and provide feedback.[1] Interpretable explanations require that a human can view the computation the machine is performing and understand exactly what is going on. This type of explanation allows the human to both trust

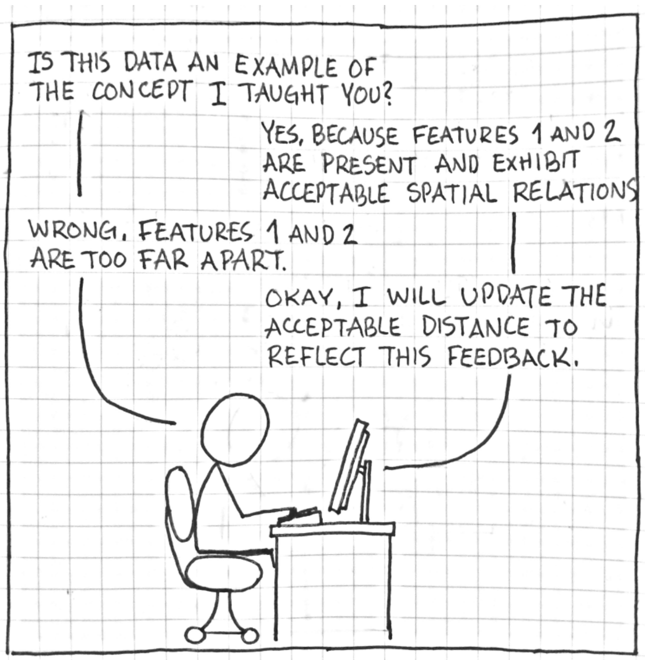


Figure 2: The big picture of where we want to go with machine learning – explainable human-machine interaction. The human gives the machine a problem to solve, the machine provides an explainable solution, it's wrong, but since the explanation is interpretable, the human is able to provide feedback, and the machine corrects its concept model. Admittedly, not as humorous.

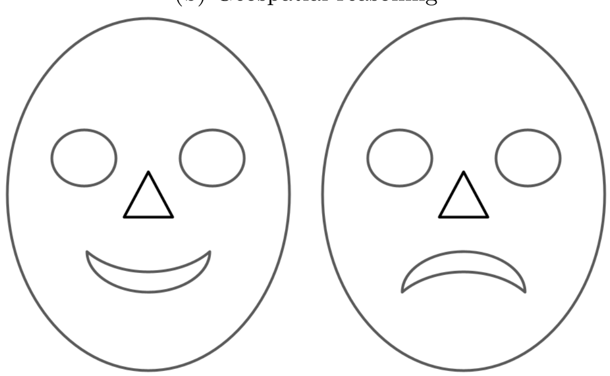
what the machine is doing and easily understand where and why the machine makes mistakes. Furthermore, actionable explanations allow the human to provide feedback to the machine by correcting the mistakes made by the machine. This is useful because there is no conversion step needed between feedback and update, maintaining the full explainability of the feedback.

Features and the relationships between features are used to distinguish exemplars of a concept from non-exemplars.[2] A machine can represent features and relationships between features as a model, and this model can be learned from data and human interaction. In Figure 2, an example dialogue between human and machine is shown. The human has pre-trained the machine to learn a concept model based on features and the relationships between features. Input data is then given to the machine, and it utilizes the learned model to determine if the data is representative of the concept or not. This determination is made based on the presence of certain features, and how well the relationships between these features match with the learned concept model. The machine's answer explains exactly what features and relationships contributed to the decision as well as the degree to which each feature and relationship contributed. Because the explanation can represent the entire chain from input to computation to output, the human can easily interpret why the machine made a certain decision.[3] Of course, this explanation could be too much information for the user; the scope of the explanation can be intelligently limited. The explanations provide the perfect interface for the human to provide actionable feedback to adjust the concept model. This feedback can adjust which features are important, how present they must be, which relationships are important, and the nature of those relationships. It is then straightforward for the machine to update its concept model to reflect the feedback provided by the human.

The process of updating the concept model lends itself well to tackling important problems related to concept drift and concept evolution.[4] There isn't always curated data to train on, leading to systems that may not generalize well, and overfit the data. Streaming data is a good way to address this problem, but leads to problems of its own. The concepts that were initially designed may need to be updated or changed to reflect the changing meaning of those concepts, and our method can allow for these changes thanks to the human feedback and update abilities.

(a) Airplane identification from 1942

(b) Geospatial reasoning



(c) Sentiment analysis

Figure 3: Some application areas for concept learning based on human-interaction and explainable AI

This method is particularly well-suited to complex scenes and objects that have regular spatial attributes. Figure 3a shows how airplane identification follows spatial patterns with changing objects. The figure details the shapes and spatial relations involved with recognizing friendly aircraft. These are very distinct spatial relationships that lend themselves to classification.[5] Figure 3b shows geospatial reasoning over complex scenes, which can be simplified to reasoning over individual parts and their spatial relationships to one another. The individual parts of a construction site can include things like work trucks, engineering vehicles, cranes, etc. These parts can then be reasoned about spatially to determine if the scene involves construction. Figure 3c shows an example of sentiment understanding. The process can be improved by tracking changing spatial relationships of individual facial features to one another. In this case, the shape of the mouth changes relative to the other features of the face, leading to a likely change in sentiment. Other applications involving visual reasoning[6] could be good use cases for our method.

3. SPECIFIC METHODS

3.1 Concept Learning

The driving force behind choosing concept learning as the paradigm to view our general problem through is the explainability benefit. Our goal here is not to completely replace pattern recognition as a technique, but rather to rely less on it for complex scenes and objects. Similarly to how complex scenes can be broken up into their constituent objects, complex objects can be broken up into their constituent features. Building a reasoning machine from simpler pattern recognition tasks allows for greater explainability. Complex pattern recognition systems, e.g. CNNs, have no specific, defined internal method for dealing with sub-features, causing these techniques to appear as a black box. A system informed by simpler parts detectors that then reasons on top of the simple parts is by-design more explainable since all of the simpler features are built into the system and easily interpretable. We call this technique concept learning because it serves as a good reference for reasoning over features to identify larger concepts/objects/scenes.[3] Simply put, we are moving the pattern recognition layer one level down from object detection to parts detection, and then performing concept learning on the parts to determine the presence of an object. Since this method is more explainable, humans can easily interpret the reasoning behind decisions made by the machine, which also provides a common language for humans to give feedback to the machine.[1]

3.2 Spatially Attributed Graph

Our main goal in building the framework of this technique is to learn concepts from features. The learning process utilizes the features themselves, and also the relationships between features[2].[7] The features and relationships are analogous to the nodes and edges of a graph, as shown in Figure 4. The features are parts detectors, and the relationships are spatial relations between parts. This natural graph structure and the usage of spatial attributes is combined into the term spatially attributed graph.[8] Here, the nodes of the graph represent the parts detectors used in the system. The edges represent every relationship and function defined between parts in the system. Formally, within the system there exists F, the set of all features. Specific features are denoted as F_i. The set of all possible relationships is denoted as R, with each unique type of relationship denoted R^t, where t is the type of relationship (distance, histogram of forces, etc.). Finally, each relationship applies to multiple features, denoted with a subscript as R^t_{ij}, where ij are all nodes that the relationship relates together, in the direction i to j if needed. In the next subsections, we discuss some of the specific parts and relationships used.

3.3 Human Interaction

A driving force behind the choice of concept learning a spatially attributed graph is the natural interaction humans can have with those specific techniques. Concept learning allows humans to give feedback to machines in order to teach them how to better recognize a concept. It is a paradigm that allows for human-in-the-loop interaction.[9] In a spatially attributed graph, the behavior of each node and edge is interpretable, allowing the human to see exactly how each node and edge contributes to the decision. Using this setup, when a machine makes a wrong decision, we can see precisely which nodes and edges don't fit with the concept definition. The language of the spatially attributed graph allows us to communicate which nodes or edges are wrong, and how to fix them to better represent the concept.

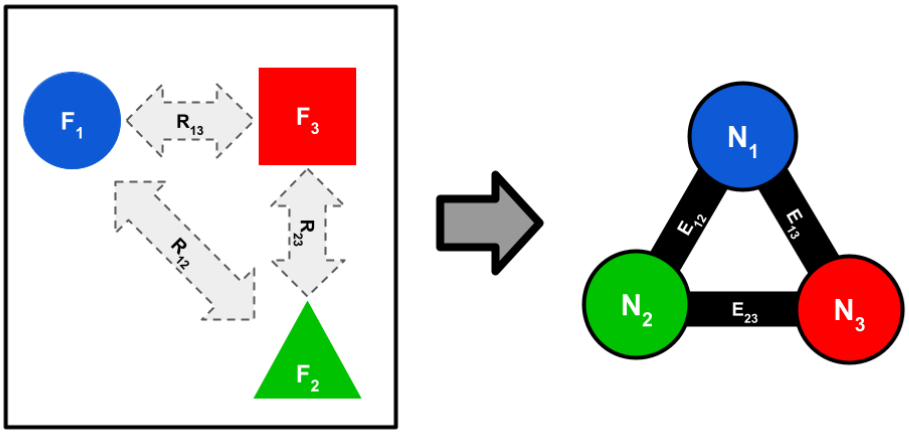


Figure 4: On the left, an image containing three features (the circle, square, and triangle). The gray arrows between the features represent the different relationships between features (distances, spatial relations, etc.).The features and relationships between features directly translate to the nodes and the edges between nodes of a graph, shown on the right.

Finally, the human in the loop has direct supervision over the learning process, since the human can provide precise feedback to directly change the feature or relationship that was incorrect. This can help prevent large errors from dominating a system, and allows for faster machine learning. This is particularly helpful for problems that require a high-degree of accuracy, yet are limited in the amount of training data they have.

3.4 Parts Detectors

Any type of parts detector can be used in our system. We don't do anything to deal with the uncertainty of the parts detectors, we treat them as a given. Currently, black-box architectures perform well at simple pattern recognition tasks.[10] We seek to utilize state-of-the-art parts detectors at the low-level, while focusing on the explainability and human interaction benefits of spatially attributed graph concept learning at the high-level. This combination allows us to use the advantages produced by state of the art parts detectors, but also not fully trust their outputs, letting us deal with the uncertainty at a higher level.

3.5 Spatial Relations

Concepts are learned from independent features, but the relationships between features provides important contextual details that allow for richer concepts to be developed by a machine. Spatial relationships are implicitly explainable, since they are interpreted by humans easily. For example, an object that exists at 90 degrees to another object could easily be interpreted as "to the right" (depending on reference frame), since spatial relationships are concepts that humans are very familiar with. Many different ways of relating objects spatially have been developed; here we focus on binary relationships. Simple relationships such as relative distances are trivial to define and can change based on context.

We use histograms of forces as one of the key tools to relate objects spatially. This type of relation calculates the force exerted by some object B on some object A, at all angles $-\pi$ to π, and creates a histogram representing the force at every angle.[11] The force at an angle relates the number of particles of object B encountered by vectors emanating from A, and the distance those particles are relative to A. As shown in Figure 5, the histogram is built from scenes where one object exerts force on another object; this force represents the direction and intensity of that force at all angles emanating outward from the argument object. This creates a histogram that is interpreted as the relative spatial positioning of the referent object to the argument object. In the case of Figure 5, it is clear that the blue blob in the top right is "above and to the right" of the green blob in the bottom left. The histogram to the right in the figure shows this as the force being most intense at and angle of $\pi/4$. The histogram is the description of the spatial relationship, and can be translated into linguistic descriptions like "above and to the right". This histogram can then inform linguistic descriptions of scenes.[11] Histograms are compared to one

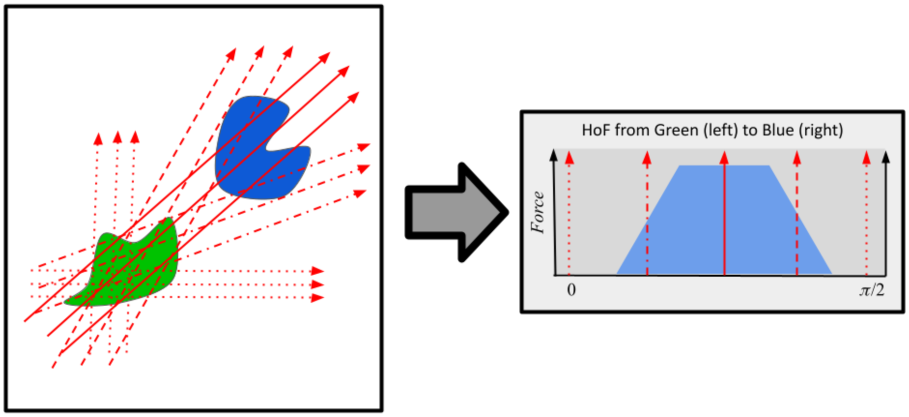


Figure 5: Two objects exist in the image on the left; the blue blob in the top right is "above and to the right" of the green blob in the bottom left of the image. The Histogram of Forces for these two objects is shown on the right.

another using similarity functions, as defined in,[12] which provides the basis for how they are used in spatially attributed graph concept learning.

4. OVERALL METHOD

4.1 Combining Features and Relationships into a Graph

For the purposes of this paper, we assume that concepts are partially constructed, and we focus on the human machine interaction. The spatially attributed graph model is composed of nodes and edges. The nodes are chosen from the set of features, or parts, that currently have parts detectors in the system. These parts detectors independently analyze the image data for their respective parts, acting as modular components that are added and removed as necessary. The edges are comprised of all relationships between two parts that are currently defined in the system. These spatial relationships are also modular, since different relations can be defined based on the problem domain. Simply, the nodes and edges come from the features found in the input data, and the relationships between features, respectively. The graph is limited by the total number of parts detectors in the system. Features that are not detectable by the system cannot be included in the graph, which is a limitation for all machine learning systems. Similarly, relationships that have not been defined are impossible to detect and add to the graph. This is to say that the possible combinations of nodes and edges are limited to what is detectable by the system (nodes), and what we have defined to be relationships between nodes (edges). This limited set of nodes and edges forms the shared "language" that machine and human can use to create and refine concepts. In the context of images in space, we refer to this as a spatially attributed graph, since the nodes and edges have spatial attributes. An example of this limited language is shown in Figure 6, where three different objects are currently detected, and all other objects in the scene are not part of the shared language of that system.

4.2 From Input to Concept Graph

This graph model can represent a concept present in an image by including features of the concept (nodes) and specific relationships between those features (edges) from the total set of features and relationships available in the system. The presence of each unique feature is determined from the set of parts detectors in the system. Using parts detectors introduces some uncertainty into the system, so the presence of each part can have a relative match strength, e.g. a number between 0 and 1 of how present that part is in the image. This presence value is used to determine which features from the total set of detectable features are present in any given input image. Once the present features are determined, each spatial relationship between features is calculated. These relationships can be of any arity, but for our purposes as defined in the previous section, we use binary relationships. These relationships, along with the features themselves, form the complete spatially attributed

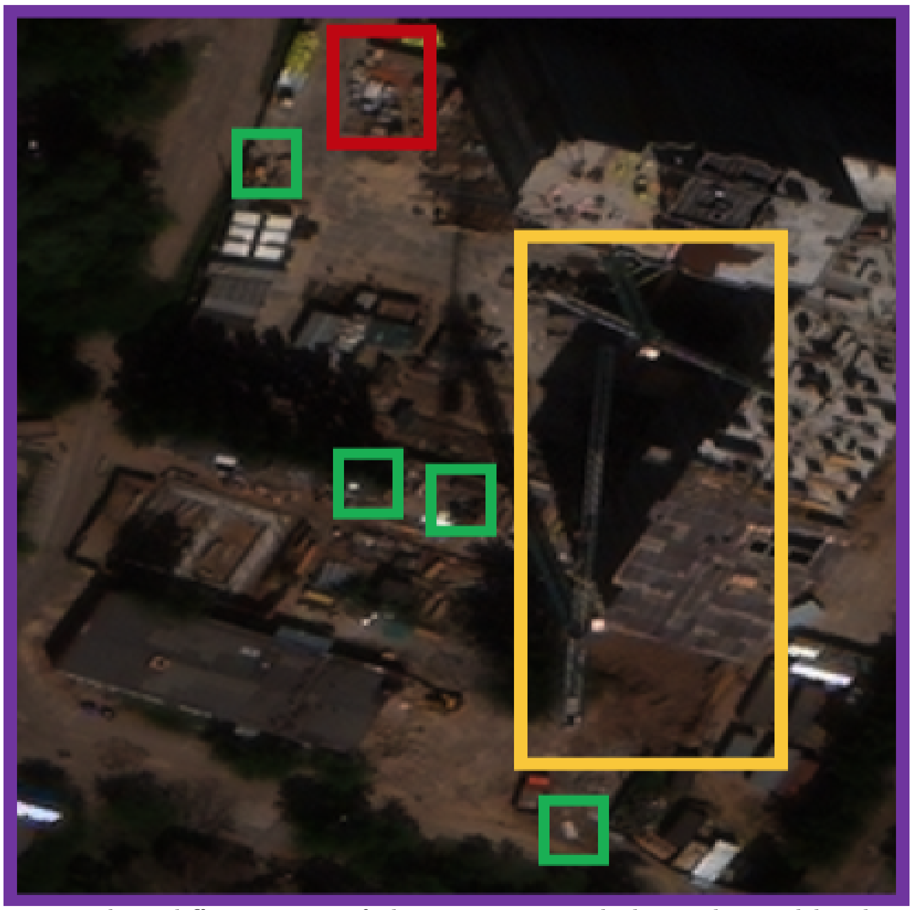


Figure 6: In this scene, three different types of objects are currently being detected by their respective parts detectors, the red box is medium trucks, the green boxes are small trucks, and the yellow box is a crane. Currently, concepts can only be built using these parts, since all other parts of the image are not being detected at this time – they are unknown to the system.

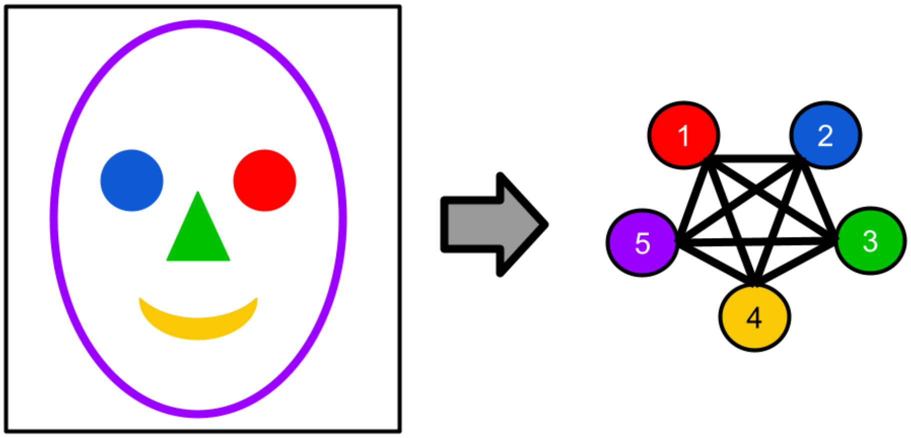


Figure 7: Each feature in the image on the left maps to a node in the graph on the right. Specifically, Node 1 = right circle, Node 2 = left circle, Node 3 = triangle, Node 4 = crescent, Node 5 = outer oval

graph for the given input image. This graph model is easily interpretable, since for each input image, every feature that was detected and the relationships between features are easy to inspect. Specifically, to demonstrate what the process would look like, we will look at the example of a cartoon face shown in Figure 7. In this Figure, each of the five features of the cartoon face in the image on the left map to nodes on the graph. This example is simple to describe, and recognizable as an example that heavily depends on the relationships between features.

4.3 Learning a Concept from Scratch

For the purposes of human-machine interaction, and for the sake of speed, a human expert can define an approximate concept using the shared language of the spatially attributed graph by selecting important features and relationships. The process of defining a concept is as simple as selecting which features must be present, and then determining the approximate spatial relations between features. The problem of learning the concept from scratch is currently unsolved, and outside of the scope of this paper. We don't discuss exactly how to learn the concept, instead we focus on comparing concepts and the human feedback loop of concept learning.

4.4 Comparing Concept Graphs

The graph model representing the input image will be referred to as the input graph, and the graph model representing the existing concept model will be referred to at the reference graph. It is easy to determine which features and relationships are necessary for a given concept by analyzing the reference graph. Each specific required feature is present in the reference graph, and the relationships between each feature are easily inspected to determine the spatial relationships between features. The input graph is compared to the reference graph to determine if the necessary concept features in the reference graph are all present in the input graph.

Using Figure 8 as an example, we can see that the reference graph on top contains 5 nodes, each specifying a specific feature. The input graph on bottom contains those same five nodes. If there are more or fewer features present in the input graph compared to the reference graph, then the concept is not present, by definition. However, extra or missing features do not necessarily mean that the input data does not demonstrate the concept or at least a partial concept, both of which can be useful. Due to the fuzzy nature of defining concepts, partial concepts may be present in images and will need to be dealt with, e.g. by forming a new concept, removing requirements from the current concept, etc. Humans can easily verify visually that the same features are present in both images. The spatial relationships between features are compared depending on the type of relationship. If the relationships are relative distances, the scalars are compared using allowable plus/minus thresholds, or a matter of degree represented by a fuzzy set. If the relationships are relative positions using histograms of forces, the histograms are compared using similarity functions. Of course, spatial relationships don't necessarily have to perfectly match. Thresholds of allowable deviation are set to allow for some variation in spatial positioning.

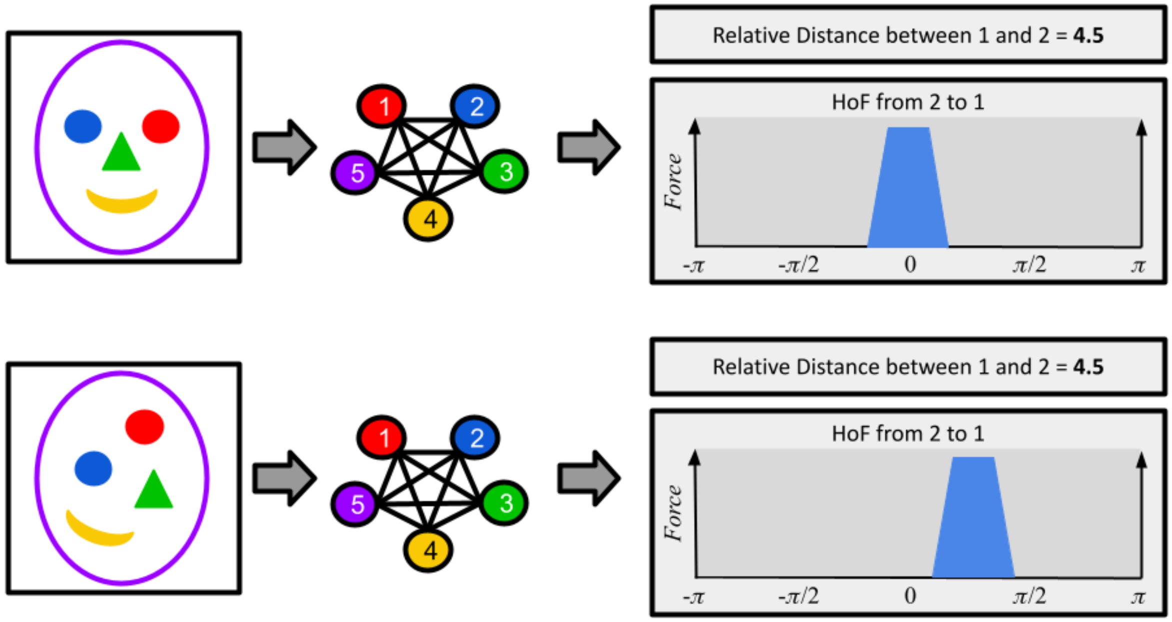


Figure 8: The comparison between input graph and reference graph. Note that while Features 1 and 2 are the same distance in both, and the histograms look similar, the histogram mean is at a different angle for the two graphs.

Depending on how each node and edge is compared, the overall graph comparison is determined using any combination function – minimum usually makes sense, because if any part of the input is wrong, the entire input doesn't fit the concept. Again, if this isn't the case, the human can provide feedback to the machine to learn a better concept. In Figure 8, the spatial relationship between Nodes 1 and 2 are shown on the right. Nodes 1 and 2 (the right eye and left eye of the cartoon face), are the same relative distance away from each other in both images, but the histogram of forces clearly shows that they are at different relative positions in the different images. The similarity between the two histograms is low, which shows that the input image does not match the reference image. This overall comparison leads to the decision of whether a concept is found within the input image or not.

4.5 Feedback Loop

For an input, the machine computes the spatially attributed graph and performs the comparison to the reference graph, deciding if the concept is present or not. Any of the set of features, relationships, and comparisons leading to the final decision can then be displayed to the human expert for evaluation, as shown in Figure 8. Of course, all of these are displayed easily, showing the full explainability of the system, but this could also be an overwhelming amount of information, given the problem domain. The human is able to analyze each step that led to the decision and determine if the decision is correct or incorrect, and either way, if any of the steps were correct or incorrect. The human can then provide feedback of varying degrees and specificity explaining what aspects are incorrect, and why, using the common language of the spatially attributed graph. An example of what this looks like is shown in Figure 9. In the example, the human gives linguistic feedback, specifically the term "farther". The machine uses this to shrink the allowable histogram of forces between the two objects to better match the intended relationship based on the term "farther". Using feedback, the machine can refine the features, relationships, thresholds, and comparisons to better represent the given concept. This process can then be repeated for more input images, and even the same input image to compare performance of concepts as they develop across time.

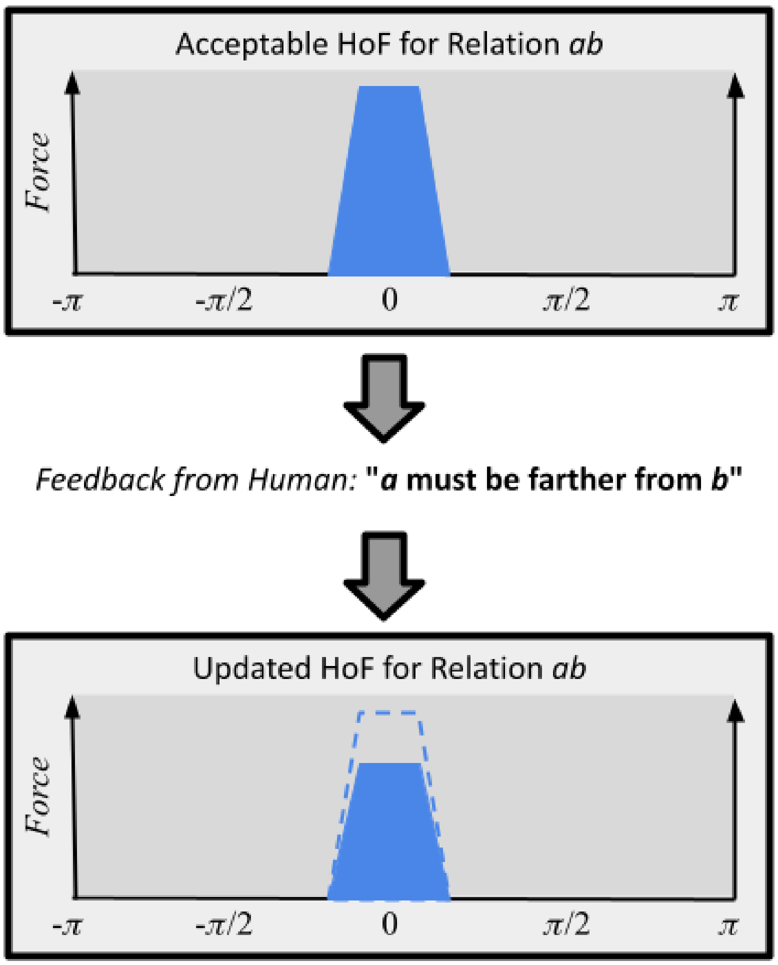


Figure 9: An example of the shared language the human can use to provide feedback to the machine to assist in concept learning. The machine has a method of translating the linguistic term "farther" into an operation on the histogram values to reduce the allowable force.

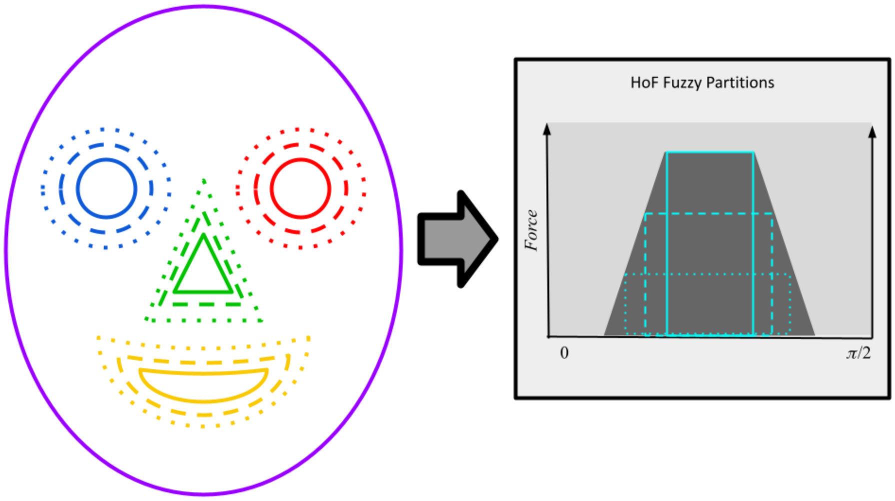


Figure 10: An example of the shared language the human can use to provide feedback to the machine to assist in concept learning.

5. EXAMPLE

5.1 Prototype

Here, we call the allowable ranges for each feature and relationship the *prototype* of the concept. This prototype represents the currently defined concept, including all relevant features, and the specifics of the relationships allowed. We use the cartoon face example because it does a good job of demonstrating why this technique is useful: specifically, because each feature is distinct and detectable, and the relationships between features defines whether or not the features represent a cartoon face or not. The prototype can be represented using crisp relationships with margins of error, or fuzzy relationships that build uncertainty into the relationship directly. The visual representation of the prototype, shown in Figure 10, is meant to show roughly where the allowable spatial positions for each feature are, based on the allowable spatial relations between features The prototype on the left represents the core of each image allowable spatial position, along with a higher and lower confidence interval to show the range of positions into which each feature could fit. The spatial relations, in effect, constrain the features to certain positions relative to one another. Of course, the figure doesn't show the relative angles or distances between individual features – for example, the right eye should be 0 degrees (to the right) of the left eye, and can't be at -2 or +2 degrees. These constraints allow the machine to determine if the image represents a certain concept or not. The flexibility in the constraints allows the machine to deal with uncertainty, as shown in the more translucent areas surrounding each crisp feature on the prototype.

5.2 Difficult False Positive

Here, we show what would happen when a false-positive is shown to the system, i.e. an input image that presents the current features and relationships of the prototype, but shouldn't, so the human has an opportunity to demonstrate the process of fixing this. In Figure 11, we see an image that currently matches the prototype for all features. The prototypical relationship between the eyes, Nodes 1 and 2, is shown in every histogram. Histogram a.) shows the allowable spatial relation between the eyes for the protoype initially. Since histogram b.), which shows the relationship between the eyes for the input image, clearly falls within the allowable region defined in histogram a.), the prototype needs to be refined to specify the relationship better. The human expert has decided that this cartoon face does not satisfy their concept of a face, so provides feedback to the machine that the right eye is too far above the left eye. The human feedback triggers and update to the prototype, which leads to the refined prototype relationship shown in histogram c.), where the relationship between the eyes has been specified to allow fewer angles.

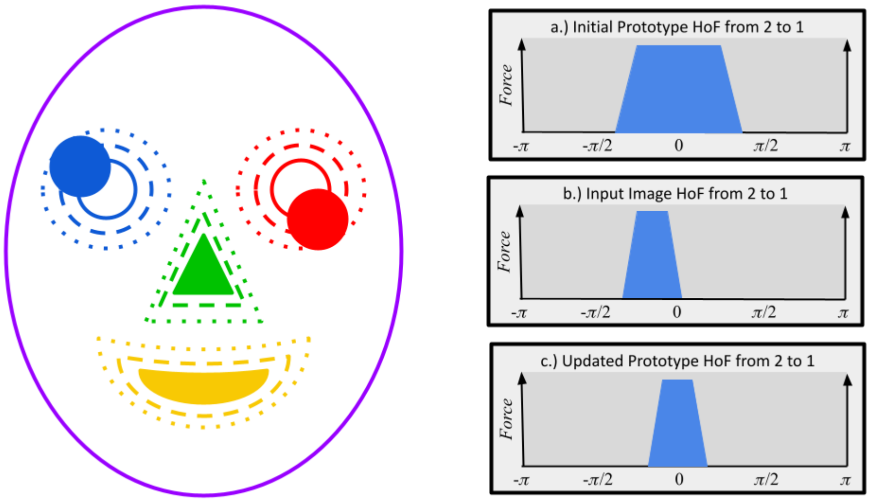


Figure 11: Input image on the left, overlaid on top of the prototype image. a.) is the prototype relationship of right eye to left eye. b.) is the input image HoF. c.) is the updated prototype after human feedback.

6. CONCLUSION

In this article, we explored concept learning based on human interaction and explainable AI. In the context of parts-based concept learning about complex scenes, we showed how features found using parts detectors act as nodes in a graph, and the spatial relations between features act as edges, creating a spatially attributed graph. We discussed the process of inputting an image into this system, and how the image data forms the spatially attributed graph. Next, we showed how concepts are represented in the spatially attributed graph, and the current state of learning a concept from data. Next, we showed how concepts are compared across different graphs due to the interpretable nature of the graphs, allowing humans to understand and take part in the process. This naturally leads to the opportunity for human interaction, where humans can improve the system using the shared language of the spatially attributed graph. Effectively, we showed how to perform online concept learning of a spatially attributed graph. We emphasized the explainability of this method by demonstrating the process using a synthetic example, helping to show the way forward towards more explainable artificial intelligence.

REFERENCES

1. H. Hagras, "Toward human-understandable, explainable ai," *Computer* **51**(9), pp. 28–36, 2018.
2. J. Gonzalez, L. B. Holder, and D. J. Cook, "Graph-based concept learning," in *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. AAAI Press*, p. 1072, The MIT Press, 2000.
3. K. Sotala, "Concept learning for safe autonomous AI," in *Artificial Intelligence and Ethics, Papers from the 2015 AAAI Workshop, Austin, Texas, USA, January 25, 2015*, T. Walsh, ed., *AAAI Workshops* **WS-15-02**, AAAI Press, 2015.
4. M. M. Masud, Q. Chen, L. Khan, C. Aggarwal, J. Gao, J. Han, and B. Thuraisingham, "Addressing concept-evolution in concept-drifting data streams," in *2010 IEEE International Conference on Data Mining*, pp. 929–934, 2010.
5. S. F. Ali, J. Jaafar, and A. S. Malik, "Proposed technique for aircraft recognition in intelligent video automatic target recognition system (ivatrs)," in *2010 International Conference on Computer Applications and Industrial Electronics*, pp. 174–179, 2010.
6. J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick, "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

7. W. Ding, B. Hu, H. Liu, X. Wang, and X. Huang, “Human posture recognition based on multiple features and rule learning,” *International Journal of Machine Learning and Cybernetics* **11**, pp. 2529–2540, Nov 2020.
8. M. A. Eshera and K. S. Fu, “An image understanding system using attributed symbolic representation and inexact graph-matching,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-8**(5), pp. 604–618, 1986.
9. J. E. van Engelen and H. H. Hoos, “A survey on semi-supervised learning,” *Machine Learning* **109**, pp. 373–440, Feb 2020.
10. L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, “Deep learning for generic object detection: A survey,” *International Journal of Computer Vision* **128**, pp. 261–318, Feb 2020.
11. P. Matsakis, J. M. Keller, L. Wendling, J. Marjamaa, and O. Sjahputera, “Linguistic description of relative positions in images,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **31**(4), pp. 573–588, 2001.
12. C. P. Pappis and N. I. Karacapilidis, “A comparative assessment of measures of similarity of fuzzy values,” *Fuzzy Sets and Systems* **56**(2), pp. 171–174, 1993.