Searching for the Most Probable Combination of Class Labels Using Etcetera Abduction

Andrew S. Gordon
Institute for Creative Technologies
University of Southern California
Los Angeles, USA
gordon@ict.usc.edu

Andrew Feng
Institute for Creative Technologies
University of Southern California
Los Angeles, USA
feng@ict.usc.edu

***Abstract*—Many machine perception tasks require a trained model to assign class labels to multiple entities in the same context, e.g., labeling multiple objects in a single photograph. In these tasks, different combinations of labels may be more likely than others, e.g., when co-occurrence biases are considered, such that the most-confident label assigned to an individual object is not always the best choice. In this paper, we propose a new method for combining evidence from multiple class probability distributions to identify the most probable combination of labels in multi-entity contexts. Our method encodes discrete class probability distributions as literals in first-order logic, and uses probability-ranked logical abduction to identify the most likely label combination, incorporating the prior and conditional probabilities of each label. We evaluate our method on two computer vision benchmarks, first for labeling common objects in photographs of everyday contexts, and second for labeling actions of athletes in sports videos. Results indicate significant gains in classifier accuracy over systems that merely select the model's most confident class label.**

***Index Terms*—I.2.6.g Machine learning I.2.3.d Inference engines I.2.4 Knowledge Representation Formalisms and Methods**

I. Introduction

In many machine learning tasks, a trained classifier is applied to multiple entities that appear in the same context. For example, a trained image classifier might be applied to multiple bounding boxes in the same photograph, or an action classifier might be applied to multiple people that appear in the same video clip. In these tasks, the classifier's most-confident label for an individual entity may not always be the best choice when considering the labels assigned to the other entities. If a potential label frequently co-occurs with another label that has been assigned to a different entity in the context, it may be a better choice than labels with higher confidence scores. Ideally, all of the known co-occurrence statistics (estimated from training data) could be exploited for all of the entities in a context, simultaneously, to identify the best combination of class labels. Doing so requires a mechanism for combinatorial search and a means of ranking combinations that incorporates the available co-occurrence information.

The project or effort depicted was or is sponsored by the U.S. Army Research Laboratory (ARL) under contract number W911NF-14-D-0005, and that the content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

In this paper, we investigate the use of probabilistic abductive reasoning as a general mechanism for identifying the most likely combination of labels in multi-entity contexts. Although logical abduction is more typically used in commonsense reasoning and diagnostic applications, our focus is on exploiting its provisions for managing a combinatorial search and for encoding statistical information in a knowledge base of logical axioms. Specifically, we investigate a variant of logical abduction called Etcetera Abduction [1], where prior and conditional probabilities (estimated from training data) can be readily encoded in first-order logical axioms. In our approach, the reasoner is provided with the discrete class confidence distributions for each entity in a single context and a knowledge base of prior and conditional probabilities, and identifies the most probable combination of labels that logically entail the input distributions.

We evaluate the effectiveness of our approach using two standard computer vision benchmarks. First, we apply our method to the assignment of class labels to bounding boxes in the Common Objects in Context (COCO) dataset of photographs in naturalistic contexts [2]. Second, we apply our method to the labeling of actions of athletes in videos of volleyball matches [3]. In both tasks, we find statistically significant gains in classification accuracy over baseline systems that simply select the most confident label for each entity.

II. Etcetera Abduction

Abductive reasoning, distinct from deductive and inductive reasoning, poses the question: *What set of assumptions best explains the observations?* Following from the early philosophical treatment by Charles Sanders Pierce, contemporary formalization of logical abduction casts it as a search for the highest-ranking set of assumptions that, when paired with a knowledge base of background axioms, logically entail a set of input observations. Abductive reasoning using propositional logic, henceforth *propositional abduction*, is defined as follows.

- **Given:** (i) Background knowledge B, (ii) observations O, (iii) set A of propositional atoms, and (iv) evaluation function $eval$, where B is a set of propositional logic formulae, and O is a set of propositional literals.

- **Find:** Among a set $\mathcal{H}$ of hypotheses, where $\mathcal{H} \equiv \{H \subseteq A \mid H \cup B \models O, H \cup B \not\models \bot\}$, find the best hypothesis $H^* \in \mathcal{H}$ that maximizes $eval(H^*)$ (i.e. $H^* = \arg\max_{H \in \mathcal{H}} eval(H)$).

Much of the research on logical abduction over the last few decades has explored various alternatives for the evaluation function $eval$, and sought efficient algorithms for abductive reasoning for more expressive logics beyond the propositional case. *Weighted Abduction* [4], for example, identifies candidate hypotheses by iteratively back-chaining from observations O that unify with the consequents of first-order logical axioms in B expressed in implicature form. Antecedent literals in these axioms are annotated with weights that translate an initial cost assigned to observation literals to entailing assumptions, allowing the $eval$ function to rank candidate hypotheses to identify the one with the least overall cost.

Etcetera Abduction [1] is a more recent variant of logical abduction that uses first-order logic and a probability-based evaluation function. In terms of probability theory, abduction can be viewed as a Maximum A Posteriori (MAP) estimation where we find the most likely hypothesis given input observations:

$$\arg\max_{H \in \mathcal{H}} eval(H) = Pr(H|O) = \frac{Pr(O|H)Pr(H)}{Pr(O)} \quad (1)$$

In abduction, this maximization problem can be simply reduced to $\arg\max_{H \in \mathcal{H}} Pr(H)$ because H logically entails O (i.e. $Pr(O|H) = 1$) and $Pr(O)$ is constant in the maximization problem. Using the same method as Poole's probabilistic Horn-clause abduction [5], Etcetera Abduction naïvely estimates $Pr(H)$ by assuming conditional independence over elemental hypotheses $h \in H$, such that the joint probability is estimated as a product of prior probabilities:

$$Pr(H) = \prod_{h \in H} Pr(h) \quad (2)$$

The key innovation of Etcetera Abduction is a mechanism for expressing defeasible first-order logical axioms in the background knowledge base B, without abandoning Poole's simple method for estimating $Pr(H)$. As an illustration of the problem that it solves, consider the following formula that might be used in a knowledge base B when interpreting a positive result when testing a person for an infectious disease:

$$(\forall p)\ (Infected(p) \rightarrow TestsPositive(p)) \quad (3)$$

When the observation O is that person $Alex$ tests positive, the set of hypotheses $\mathcal{H}$ will include the hypothesis $Infected(Alex)$, which fully entails the observations O, and whose probability is equal to the prior probability of this one literal. However, this probability estimate fails to incorporate the conditional probability of testing positive given that $Alex$ is infected, i.e., the *true positive rate* of the diagnostic test. Formula (3) is *defeasible*, in that it is not always true, and cannot be included as an axiom in our knowledge base B.

Etcetera Abduction expands on the solution originally used in Weighted Abduction [4]. A defeasible formula such as (3) is made into a (non-defeasible) axiom by including a special literal as an additional conjunct in the antecedent, an *etcetera literal*, meant as a proxy for all of the uncertainty that would also have to be assumed in order for the antecedent to logically entail the consequent. To be included as an axiom in knowledge base B, formula (3) would be rewritten as follows:

$$(\forall p)\ (Infected(p) \wedge Etc_{42}(0.94, p) \rightarrow TestsPositive(p)) \quad (4)$$

The literal $Etc_{42}(0.94, p)$ represents the innumerable antecedents that must also be true for a person infected with the disease to receive a positive test result, e.g., the test must have been administered correctly to the person, the reactive chemicals in the test have been correctly manufactured, the person's sample contains no agents that would interfere with the chemical reaction, etc.

Formally, an etcetera literal E for a definite clause $A \wedge E \rightarrow C$ is defined as a disjunction of all possible conjunctions e where $A \wedge e \rightarrow C$, such that $A \wedge C \rightarrow E$. Or informally, whenever we have both A and C, the other conditions must have also been right for them both to be true.

In this definition, $(A \wedge C)$ is true exactly when $(A \wedge E)$ is true, giving us a means of specifying a probabilistic semantics for etcetera literals. Where truth values represent the occurrence or nonoccurence of events, we have this equality between their joint probabilities.

$$Pr(A, E) = Pr(A, C) \quad (5)$$

As with all literals in probabilistic Horn-clause abduction [5], we assume etcetera literals are conditionally independent from all other literals, such that:

$$Pr(A)Pr(E) = Pr(A, C) \quad (6)$$

Solving for $Pr(E)$ gives us a conditional probability:

$$Pr(E) = Pr(C|A) \quad (7)$$

That is, the prior probability of an etcetera literal is equal to the conditional probability of the consequent given the rest of the antecedent. In axiom (4), the prior probability of the etcetera literal is equal to the true positive rate of the test:

$$Pr(Etc_{42}(0.94, p)) = Pr(TestsPositive(p) \mid Infected(p)) \quad (8)$$

The subscript in an etcetera literal's predicate symbol, e.g., Etc_{42}, uniquely identifies the uncertainties that are specific to a given axiom and all of its quantified variables. That is, it will only appear in one axiom in knowledge base B. By convention, the prior probability of each etcetera literal is included as its first argument as a numerical constant (0.94 in axiom (4)), followed by all other axiom variables.

When an etcetera literal E is the only antecedent for a consequent C, as in the axiom $E \rightarrow C$, then the prior probability of the etcetera literal E is equal to the prior probability of the consequent C. This affords a simple means

of encoding prior probabilities for all elemental hypotheses. For example, the base rate of infection for a particular disease can be encoded in its own etcetera literal:

$$(\forall p)\ (Etc_{53}(0.035, p) \rightarrow Infected(p)) \tag{9}$$

Software implementations of Etcetera Abduction conduct their search for hypotheses $\mathcal{H}$ by iteratively back-chaining from observations O using background knowledge B, accepting a hypothesis H when all conjuncts $h \in H$ are etcetera literals, and $Pr(H)$ is the product of the numerical constants encoded as the first arguments in each literal. For example, in a knowledge base B that includes axioms (4) and (8), the observation $O = TestsPositive(Alex)$ is entailed by the hypothesis $H = Etc_{53}(0.035, Alex) \wedge Etc_{42}(0.94, Alex)$, such that $Pr(H) = 0.035 \times 0.94$.

Previous work has explored the application of Etcetera Abduction to a variety of commonsense interpretation problems [1], [6], employing dozens or hundreds of knowledge base axioms. As the size of the observations O and the background knowledge B grows in size, the combinatorial search for hypotheses $\mathcal{H}$ becomes intractable, requiring incremental approaches that sacrifice a guarantee of optimal solutions for tractable search [7].

III. Approach

In this paper, we propose a new method for selecting the best combination of labels for entities in the same context, by encoding confidence, prior, and co-occurrence probabilities in first-order logic, and using Etcetera Abduction to conduct the combinatorial search for the most likely label set. This section describes the three main components of our approach to processing the output of a machine learning model applied to multiple entities in the same context. First, we encode the model's class confidence distribution for each entity in a context as a literal in first-order logic. Second, we use available training data to estimate prior and conditional probabilities of labels, and encode these estimates in definite clauses in first-order logic. Third, we use Etcetera Abduction to conduct a combinatorial search for the most probable label combination, and evaluate the accuracy of this selection for each individual entity.

A. Encoding confidence distributions as literals

Our method begins by encoding a model's confidence in label assignments as a literal in first-order logic. Nearly every machine learning method for multi-class classification is able to output confidence values for all possible classes when processing a given test input, although only the most-confidence class is typically used in accuracy evaluations. Without some care, however, the output confidence distributions of models are often not well-calibrated, i.e., the expected sample accuracy does not directly correspond to the model's confidence. In contemporary neural network classifiers that incorporate a final softmax layer to produce confidence distributions, *temperature scaling* has been used as an effective calibration method, where the output entropy is raised without changing the classifier's accuracy [8] to produce a vector that better approximates the class probability distribution.

Given a class probability distribution for a single entity in a multi-entity context, we select the top N most-probable labels ($N = 4$ in all of our experiments), and encode both the labels and the probabilities as arguments (constants) in a single literal, as follows:

$$Top4(classifierID, sampleID, class1, pr1, class2, pr2, class3, pr3, class4, pr4) \tag{10}$$

Here, $classifierID$, $sampleID$, $class1$, $class2$, $class3$, and $class4$ are all represented as string constants, while the corresponding probabilities $pr1$, $pr2$, and $pr3$, and $pr4$ are encoded as numerical (floating-point) constants between 0 and 1. For example, the following literal encodes the top four most confident labels from a well-calibrated image classifier:

$$Top4(\text{“}\texttt{ResNet50}\text{”}, \text{“}\texttt{test32}\text{”}, \text{“}\texttt{tick}\text{”}, 0.5331, \text{“}\texttt{snail}\text{”}, 0.0383, \text{“}\texttt{slug}\text{”}, 0.0195, \text{“}\texttt{fly}\text{”}, 0.0129) \tag{11}$$

During the search for the most-probable label combination, each of the labels encoded in this literal is treated as a candidate, with the corresponding label probability encoded in the form of an etcetera literal. To unpack the labels and probabilities from the encoding, we utilize a fixed set of axioms—one for each of the N encoded possible labels—to select a label and assign its likelihood to an etcetera literal. For example, where $N = 4$, the following axiom is used to make the assumption that the third label is the correct one:

$$\begin{aligned}(\forall\ & classifierID, sampleID, \\ & class1, pr1, class2, pr2, class3, pr3, class4, pr4) \\ & Class(classifierID, sampleId, class3)\ \wedge \\ & Etc_3(pr3, classifierID, sampleID, class1, pr1, \\ & class2, pr2, class3, pr3, class4, pr4) \\ & \rightarrow Top4(classifierID, sampleID, class1, pr1, \\ & class2, pr2, class3, pr3, class4, pr4)\end{aligned} \tag{12}$$

During the search process, the literals that encode the confidence distribution unify with consequents of these N axioms, yielding two new assumptions that logically entail the confidence literal. For the encoding in (11) above, these two assumptions are as follows:

$$\begin{aligned}& Class(\text{“}\texttt{ResNet50}\text{”}, \text{“}\texttt{test32}\text{”}, \text{“}\texttt{slug}\text{”})\ \wedge \\ & Etc_3(0.0195, \text{“}\texttt{ResNet50}\text{”}, \text{“}\texttt{test32}\text{”}, \text{“}\texttt{tick}\text{”}, 0.5331, \\ & \text{“}\texttt{snail}\text{”}, 0.0383, \text{“}\texttt{slug}\text{”}, 0.0195, \text{“}\texttt{fly}\text{”}, 0.0129)\end{aligned} \tag{13}$$

Here the $Class$ literal is left to be further explained by back-chaining on additional knowledge base axioms (see below), while Etc_3 becomes a factor in a hypothesis's naïve estimate of joint probability, where $Pr(Etc_3) = 0.0195$).

B. Encoding prior and co-occurrence probabilities as axioms

In this work, we consider two types of assumptions that logically entail a class assignment, e.g., the $Class$ literal in (13), above. First, the assignment could be fully entailed by its prior probability, which could be estimated from any available training data. For example, the following axiom encodes the prior probability of the class label in (13), where an etcetera literal entails the class assignment:

$$\begin{aligned}&(\forall\ classifierID, sampleID)\\ &\quad Etc_{slug}(0.0007, classifierID, sampleID)\\ &\quad \rightarrow Class(classifierID, sampleID, \text{“}\texttt{slug}\text{”}) \end{aligned} \quad (14)$$

Alternatively, the label assignment could be fully entailed by its co-occurrence with a particular label assigned to a different entity in the same context, with the co-occurrence probability again estimated from any available training data. For example, the following axiom encodes the conditional probability of the label “`slug`” given that the label “`leaf`” was assigned to a different sample in the same context:

$$\begin{aligned}&(\forall\ classifierID, sampleID, otherID)\\ &\quad Class(classifierID, otherID, \text{“}\texttt{leaf}\text{”}) \wedge\\ &Etc_{slug|leaf}(0.2702, classifierID, sampleID, otherID)\\ &\quad \rightarrow Class(classifierID, sampleID, \text{“}\texttt{slug}\text{”}) \end{aligned} \quad (15)$$

In our experiments, we automatically generate a knowledge base consisting of both of these types of axioms by analyzing available training data.

C. Search for the most probable label combination

We use an open-source implementation of incremental Etcetera Abduction [7] to identify the most probable combination of label assignments for multiple entities appearing in the same context. The top N most-probable label assignments for entities are encoded as literals, one literal for each entity in the context, and passed as observations O to the reasoner, along with the top N axioms and (automatically generated) probability axioms as the background knowledge base B. The reasoner identifies the most-probable hypothesis H from candidates $\mathcal{H}$, and the predicted label for each entity is extracted from the $Class$ literals that are entailed by H.

IV. Evaluation 1: Object Classification

We conducted our first set of experiments using the popular Common Objects in Context (COCO) dataset [2], which consists of photographs in everyday settings containing multiple, annotated objects. This benchmark for object detection and classification consists of photographs with bounding boxes, instance masks, and keypoint annotations (see Figure 1). In our experiments, we utilized the object detection annotations from the 118K images in the training split and 5K images in the validation split, where objects are classified into one of 80 object classes.

Our motivation for using the COCO dataset stemmed, in part, from our analysis of the co-occurrence of object classes in these photographs. We found that 76.6% of gold-standard class label assignments in the COCO training data split occurred in photographs where there was at least one other object with the same class label assignment, and that the likelihood of co-occurrence varied greatly among classes. For example, *sheep*, *carrot*, and *book* very frequently co-occurred, while *toaster*, *hair drier*, and *fire hydrant* co-occurred very infrequently. In our experiments with the COCO dataset, we focused specifically this likelihood of co-occurrence. Using the COCO training data split, we computing both the prior and co-occurrence probabilities for each of the 80 object classes, and encoded these values in knowledge base axioms as etcetera literals, as described in the previous section.

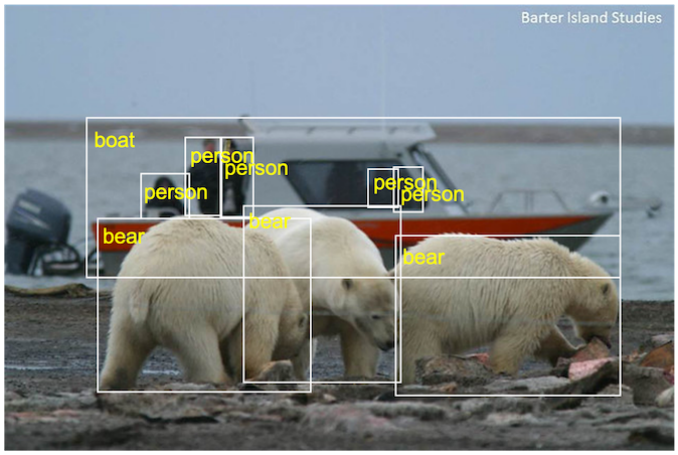

Fig. 1. An example annotated photograph from the COCO dataset showing gold-standard class labels.

To generate class confidence distributions for each object in the the validation split, we trained a standard ResNet-50 classifier using the COCO training data split. Each bounding box in the dataset was resized to a resolution of 224x224 pixels, with 859k samples using for training, and 36k samples in the validation split. The trained model was applied to each object in the validation split, and the confidence values for the top four class labels for each object were encoded as first-order logical literals, as described in the previous section. The set of confidence literals for each photograph were used as input observations to an implementation of incremental Etcetera Abduction [7], with search parameters of $depth = 3$, $window = 3$, and $beam = 3$. The class labels for each object were extracted from the most-probable solution, and compared against the baseline of selecting the most confidence class. Statistical significance (p-values) between the results of different approaches was computed using stratified shuffling [9].

The results in Table I show that our approach achieves significant and substantial gains in accuracy over the baseline, with improvements of over five percentage points.

To better understand the relative contribution of the conditional and prior probabilities in improving the accuracy, we conducted an ablation study where the conditional probability

TABLE I
OBJECT CLASSIFICATION IN THE COCO DATASET

Approach	Accuracy[a]	Gain
trained classifier, confidence only (baseline)	69.22%	
trained classifier, most-probable combination	**74.65%**	+5.43%
trained classifier, priors-only (ablation)	72.40%	+3.18%
CLIP classifier, confidence only (baseline)	47.85%	
CLIP classifier, most-probable combination	**53.77%**	+5.92%

[a]All differences significant at $p < .001$

axioms were removed from the knowledge base. This is equivalent to re-weighting the confidence scores by the prior probability of each class label. As seen in Table I, more than three percentage points of gains in accuracy can be achieved using only these prior probabilities.

We also investigated whether comparable improvements could be achieved when using zero-shot CLIP-based classifier [10], not trained on the COCO training data split. Using text labels corresponding to the 80 object classes in the COCO dataset, we applied a CLIP-based classifier to each of the bounding boxes in the validation split, and again encoded class confidence values as input literals to Incremental Etcetera Abduction, along with the knowledge base of prior and co-occurrence axioms. As seen in Table I, we observed almost six percent of gain in accuracy compared to the baseline of selecting the most confident class.

V. EVALUATION 2: ACTION RECOGNITION

We conducted a second set of experiments for the task of action recognition in video, using an annotated dataset of volleyball matches [3]. Developed in support of research on group activity recognition, this dataset contains 4830 videos annotated with a single group activity class label among eight possible labels (see Figure 2). In addition, each of twelve volleyball players are individually annotated with bounding boxes and an action label, from a set of ten possible action class labels, e.g., *standing* and *blocking*, including a *none* label when a player is out-of-view or the action was otherwise not annotated. For our experiments, only the individual action class labels were used, with *none* labels removed in our experiments.

To generate class confidence distributions for the action assigned to each individual player, we trained an action recognition model using the 27K samples in the training data split (3493 videos). The recognition model is based on C3D network architecture [11], which uses 3D convolutional layers to handle time dimension for action recognition. Instead of using 2D image patches as input, the skeletal keypoints extracted using HRNet [12] were used as input features for training the recognition model. The network was trained for 230 epochs using SGD optimizer with initial learning rate 0.4 and Cosine Annealing LR scheduler. Since the dataset is heavily biased toward one class label (the *standing* action), directly training the model on all samples tend to overfit and could produce a trivial model that always predict a single label. To alleviate this issue, we oversampled the data from

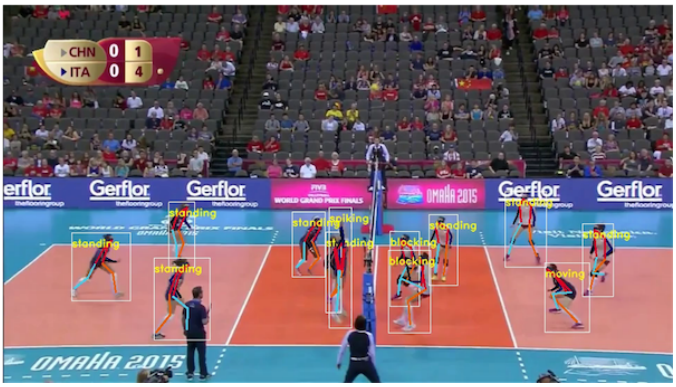

Fig. 2. An example annotated video frame from Volleyball dataset showing gold-standard class labels. Most-probable class labels are computed for each team (six players) independently.

the less frequent classes to ensure a balanced dataset during training. To prevent overfitting due to oversampling data, we also applied data augmentations by randomly resizing and horizontally flipping the key point poses. The resulting baseline model has 79.99% top-1 classification accuracy and 62.01% mean class accuracy.

With the goal finding the most-probable combination of individual action labels within a context, we procedurally split the 12 players within a video into two teams and treated each team as a separate context. Specifically, the bounding boxes of the players were first sorted based on the X-coordinates. Then the first 6 entries were selected as the Left team while the last 6 entries were the Right team, creating two distinct contexts for each video in the training and test splits.

The knowledge base of prior and conditional probabilities in our experiment was generated automatically by analyzing the contexts in the training data split (two for each video). We encoded prior probabilities (9 axioms) and pairwise conditional probabilities for the nine action class labels (81 axioms).

Our evaluation compared our method to the baseline approach of selecting the most confident action class label for each player. Within each team context, the confidence distribution for action labels for each player was first inferred using our trained action recognition model, then the most-probable combination of action labels was identified using incremental Etcetera Abduction, with search parameters of $depth = 3$, $window = 3$, and $beam = 3$. Statistical significance of observed differences was computed using stratified shuffling.

The results in Table II show that the application of our method yields a small but statistically significant improvement of one percent in classification accuracy. This improvement was not as substantial as seen in the object classification experiments. Our intuition is that this is due, in part, to the diversity of contexts that are exhibited across the two tasks. In the COCO experiments, the huge diversity of photograph contexts, large number of object class labels, and varied number of entities increases the importance of the contextual information encoded in the knowledge base. In the volleyball dataset, with only nine individual action classes, fixed-sized

teams, and a narrow scope of group activity across the dataset, the relative importance of contextual information may be much lower.

TABLE II
ACTION RECOGNITION IN THE VOLLEYBALL DATASET

Approach	Accuracy[a]	Gain
trained classifier, confidence only (baseline)	79.99%	
trained classifier, most-probable combination	**81.00%**	+1.02%

[a]Difference significant at $p < .001$

VI. Related Work

Instead of inferencing a single label, multi-label classification predicts multiple attributes or class labels from a global context such as an image. Its main challenge lies in handling the dependency between different classes or objects. Such label dependency could be modelled via probablistic graphical models by learning the condtional label structure [13] or by constructing auxiliary labels from informative label combinations [14]. Since such label relationships naturally forms a dependency graph, recent works applied graph convolutional networks (GCN) to learn a feature representation that captures the label correlations between objects [15], [16]. Different from our proposed method, these previous works developed end-to-end model that learn the label correlations as part of the classification model. On the other hand, our method is intended as a post-processing module that would work with a pre-trained model that only handles individual labels. Thus it is able to re-purpose an existing model for predicting better labels given a global context.

VII. Discussion

Logical abduction is typically used in knowledge-rich tasks, such as commonsense reasoning, language understanding, and diagnosis, where carefully-crafted knowledge bases encode relevant domain information. However, in this paper we use logical abduction primarily as a mechanism for combinatorial search. Our specific choice of abductive reasoning, Etcetera Abduction, allows for a Maximum A Posteriori (MAP) estimation where both prior and conditional probabilities are encoded as axioms in first-order logic, automatically generated from statistical analysis of training data splits. Viewed as an interpretation problem, we are interpreting the uncertain output of a trained classifier, searching for the best explanation (combination of class labels) given what we know (prior and conditional probabilities).

Other opportunities afforded by logical abduction remain to be explored in future work. In particular, the relational nature of first-order logic allows for the expression of information about the relationships between entities and their structural roles in the overall context. Alternative formulations of the knowledge bases used in our experiments could support inference about the joint activity of *setting* and *spiking* between two players in the volleyball domain, or inference about the role of the boat in separating people from bears in Figure 1. Similarly, our approach affords the easy integration of external information that might be known about the context itself that might influence its interpretation, e.g., that the photograph in Figure 1 was itself taken from a boat, or that the match in Figure 2 was the Olympic finals.

Among the unanswered research questions is whether these capabilities for combinatorial search and relational reasoning can be implemented in the same neural network frameworks as the upstream classification models. Our experiments demonstrate that doing so would improve classifier accuracy when applied in multi-entity contexts, and our approach highlights the key functionalities that would need to be approximated in future neural network architectures.

References

[1] A. S. Gordon, "Commonsense Interpretation of Triangle Behavior," in *Thirtieth AAAI Conference on Artificial Intelligence*. Palo Alto, CA: AAAI Press, 2016, pp. 3719–3725.

[2] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[3] M. S. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori, "A hierarchical deep temporal model for group activity recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1971–1980.

[4] J. R. Hobbs, M. E. Stickel, D. E. Appelt, and P. Martin, "Interpretation As Abduction," *Artificial Intelligence*, vol. 63, no. 1-2, pp. 69–142, Oct. 1993.

[5] D. Poole, "Probabilistic Horn Abduction and Bayesian Networks," *Artificial Intelligence*, vol. 64, no. 1, pp. 81–129, 1993.

[6] A. S. Gordon and U. Spierling, "Playing Story Creation Games with Logical Abduction," in *International Conference on Interactive Digital Storytelling*. Springer, 2018, pp. 478–482.

[7] A. S. Gordon, "Interpretation of the Heider-Simmel Film using Incremental Etcetera Abduction," *Advances in Cognitive Systems*, vol. 6, pp. 1–16, 2018.

[8] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 1321–1330.

[9] E. W. Noreen, *Computer intensive methods for hypothesis testing: An introduction*. New York: Wiley, 1989.

[10] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 8748–8763.

[11] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, "Revisiting skeleton-based action recognition," *arXiv preprint arXiv:2104.13586*, 2021.

[12] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *CVPR*, 2019.

[13] Q. Li, M. Qiao, W. Bian, and D. Tao, "Conditional graphical lasso for multi-label image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2977–2986.

[14] X. Li, F. Zhao, and Y. Guo, "Multi-label image classification with a probabilistic label enhancement model." in *UAI*, vol. 1, no. 2, 2014, pp. 1–10.

[15] Z.-M. Chen, X.-S. Wei, P. Wang, and Y. Guo, "Multi-label image recognition with graph convolutional networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5177–5186.

[16] R. You, Z. Guo, L. Cui, X. Long, Y. Bao, and S. Wen, "Cross-modality attention with semantic graph embedding for multi-label classification," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 12 709–12 716.