Evaluating and Improving Interactions with Hazy Oracles

Stephan J. Lemmer and Jason J. Corso

University of Michigan, Ann Arbor, Michigan, USA
lemmersj@umich.edu, jjcorso@umich.edu

Abstract

Many AI systems integrate sensor inputs, world knowledge, and human-provided information to perform inference. While such systems often treat the human input as flawless, humans are better thought of as *hazy oracles* whose input may be ambiguous or outside of the AI system's understanding. In such situations it makes sense for the AI system to defer its inference while it disambiguates the human-provided information by, for example, asking the human to rephrase the query. Though this approach has been considered in the past, current work is typically limited to application-specific methods and non-standardized human experiments. We instead introduce and formalize a general notion of deferred inference. Using this formulation, we then propose a novel evaluation centered around the Deferred Error Volume (DEV) metric, which explicitly considers the tradeoff between error reduction and the additional human effort required to achieve it. We demonstrate this new formalization and an innovative deferred inference method on the disparate tasks of Single-Target Video Object Tracking and Referring Expression Comprehension, ultimately reducing error by up to 48% without any change to the underlying model or its parameters.

Introduction

Many artificial intelligence systems are motivated by intuitive interaction with humans: by combining sensor inputs, world knowledge, and human-provided information, they perform useful inferences such as answering visual questions (Antol et al. 2015), propagating an initial segmentation through a video (Perazzi et al. 2016), or integrating semantic information to resolve perceptual ambiguity (Szeto and Corso 2017). Despite the well-understood ambiguities and difficulties of human-provided information explored in the human-computer interaction domain (Ipeirotis, Provost, and Wang 2010; Bhattacharya, Li, and Gurari 2019; Gordon et al. 2021), state-of-the-art works in machine learning generally treat the human as an oracle that provides a single piece of flawless information then disappears. While this formulation simplifies dataset-based evaluation, we believe it more appropriate to treat humans as *hazy oracles*: the information they provide may be incorrect, ambiguous, or outside of the system's understanding of the world, but they can provide clarifying information after the initial query.

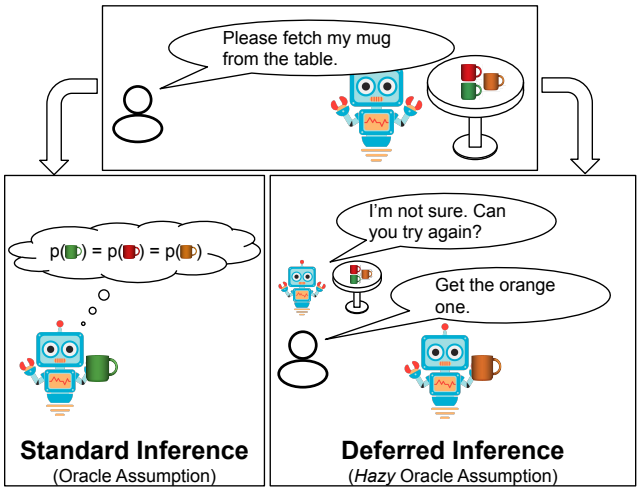

Figure 1: A simple example of the benefit of our approach. The human provides an ambiguous request that standard inference will misinterpret two-thirds of the time. By assuming a hazy oracle, the robot is able to defer the inference and obtain additional information to solve the task. Robot image via Wikimedia commons (CC-SA), mug images via freepik.com. Best viewed in color.

For example, consider the scenario shown in Figure 1: a robot must perform the task of retrieving an orange mug given a verbal command by a user. While the robot is likely to misunderstand *please fetch my mug from the table* if it does not know which mug belongs to the speaker, it can accomplish the desired task by delaying its response and requesting more information from the human. While conceptually simple, this mechanism of *deferred inference* is a challenge in practice: the robot must determine whether to defer and, if it chooses to defer, determine how best to integrate the potentially noisy additional information.

Although the benefit of deferred inference is intuitive both for individual (Gurari et al. 2018) and crowd (Lemmer, Song, and Corso 2021) interactions, no standard approach for implementation or evaluation has emerged. Selective prediction approaches (Chow 1970; Geifman and El-Yaniv 2017) evaluate performance thoroughly but do not

consider subsequent inputs for the same task, and methods that defer inference (Mees and Burgard 2020; Sharma et al. 2022) typically validate their methods via small-scale problem-specific human experiments, limiting dataset size and comparability between evaluations. Additionally, since such studies require deferral criteria to be set prior to the experiment, the complex interplay between accuracy and burden on the user is only described at one point. This is a critical oversight, as the best deferral method often changes based on how many deferrals are permitted (Lemmer, Song, and Corso 2021; Lemmer and Corso 2021).

To appropriately facilitate interaction with hazy oracles, we therefore introduce a more principled formulation that addresses these limitations and enables low-cost direct-comparison between deferred inference methods. To do this, we propose a new metric for measuring performance—Deferred Error Volume (DEV)—that explicitly and thoroughly evaluates the tradeoff between inference error and the additional human effort required to achieve it. Alongside this evaluation method, we introduce a method for deferred inference based on a belief update. While simple, our proposed method is both generalizable and remarkably effective: it outperforms previous methods by more effectively integrating additional information and reducing the likelihood of multiple deferrals, and it can be easily applied to novel applications and architectures.

We demonstrate this versatility by implementing our evaluation and method on the disparate applications of Single-Target Video Object Tracking (Kristan et al. 2016) and Referring Expression Comprehension (Mao et al. 2016). On both applications, we show significant improvement over both deferral-free inference and deferral methods proposed in other works, reducing error by up to 48% under an acceptable level of human effort with no modifications to the training or architecture of the underlying task model.

Our main contributions are as follows:

- A general formulation and evaluation method for deferred inference with hazy oracles, centered around the Deferred Error Volume (DEV) metric, that provides a fair and thorough comparison between methods.
- A deferred inference method that significantly improves performance over both the base model without deferral and deferral baselines based on previous work.
- An Evaluation of our method on the disparate tasks of Visual Object Tracking and Referring Expression Comprehension that shows the need for a novel evaluation, the generalizability of our evaluation and solution, and the quantitative benefit—up to a 48% reduction in error—of our proposed method.

Related Work

Aggregating Human Inputs Many works, particularly in the crowdsourcing domain, use multiple human inputs to increase accuracy. Though some works (Branson et al. 2010; Russakovsky, Li, and Li 2015) allow the model to choose when to terminate, the most common approach is to allow the human operator to review the model's output directly and provide new information until the result is satisfactory (Jain and Grauman 2016; Gouravajhala et al. 2018; Choi et al. 2019; Agustsson, Uijlings, and Ferrari 2019; Uijlings, Andriluka, and Ferrari 2020). These approaches are sufficient for dataset collections: performing tasks such as answering questions about given bounding boxes (Russakovsky, Li, and Li 2015) or confirming answers (Uijlings et al. 2018) is faster and more accurate than generating the dataset through drawing a bounding box directly on the image. However, such methods encode the assumption that one can interact directly in the output space (*e.g.,* you are capable of understanding and manipulating the image). This assumption is impractical in important cases: if a visual question answering system is assisting a visually impaired individual (Gurari et al. 2018), that individual can not manually confirm if the answer is correct, but can easily rephrase the question.

Deferred Inference Deferred inference can be meaningfully applied to many applications (Antol et al. 2015; Das et al. 2017; Szeto and Corso 2017; Anderson et al. 2018; Perazzi et al. 2016), but the majority of existing works focus on Visual Question Answering (Mahendru et al. 2017; Uehara, Duan, and Harada 2022) or Referring Expression Comprehension (Nyga et al. 2018; Shridhar and Hsu 2018; Mees and Burgard 2020; Sharma et al. 2022), likely due to the intuition of asking for more information during conversation. Such methods—which often rely on a difficult text-generation task—require some combination of restrictive assumptions, more complex architectures, novel datasets, and human experimentation. Two works (Hatori et al. 2018; Lemmer, Song, and Corso 2021) follow our approach and mitigate these challenges by receiving a deferral response in the same format as the task definition. These works serve as baselines for both our evaluation and proposed deferral method.

Evaluation Methods Most works related to deferred inference set deferral conditions prior to their experiments and report the final error. When user burden is specified, it is most often a point metric corresponding to the reported accuracy, such as time-per-annotation (Agustsson, Uijlings, and Ferrari 2019; Uijlings, Andriluka, and Ferrari 2020) or number of annotations per target (Ipeirotis, Provost, and Wang 2010; Hatori et al. 2018). The value of these point measurements is questionable: when a deferral method is evaluated at different thresholds, the best method often changes (Lemmer and Corso 2021; Lemmer, Song, and Corso 2021).

Works in selective prediction (Chow 1970; Cortes, DeSalvo, and Mohri 2016; Geifman, Uziel, and El-Yaniv 2019; Yildirim, Ozer, and Davulcu 2019; Mozannar and Sontag 2020) often evaluate a relevant tradeoff between the proportion of classifications performed and the error, but implicitly or explicitly assume the human can provide the correct answer—which can not be guaranteed in settings such as crowdsourcing (Lemmer, Song, and Corso 2021) or public-facing deployments (Bhattacharya, Li, and Gurari 2019). An exception to this is the work of Bondi *et al.* (Bondi et al. 2022), which acknowledges the possibility of human failure after deferral, but, like crowdsourcing works, requires the human to directly perform the task.

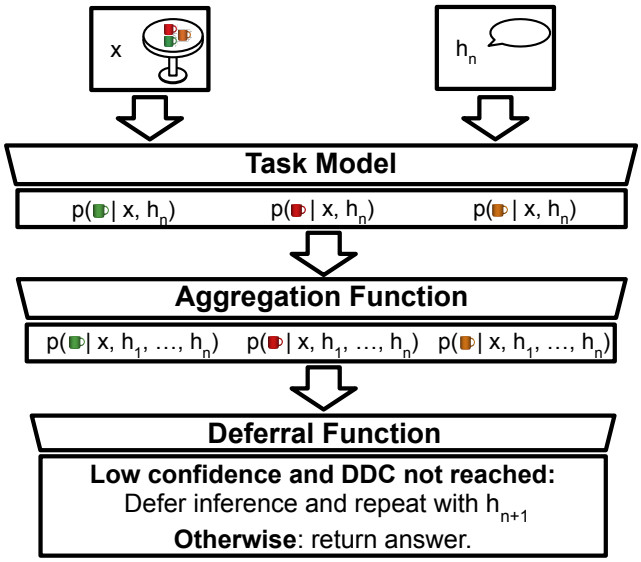

Figure 2: Our formulation of deferred inference, with the task shown in Figure 1 used for illustration. Here, x is the robot's perception and h_n is the human input at step n.

Problem Statement

An automated agent is asked to perform a task, such as cropping an image based on a language request or tracking an object through a video. This agent has some probability of solving the task on its own, but may also defer to a *hazy oracle* that can provide additional information at some cost. The information provided by the hazy oracle may be ambiguous either in truth (*e.g.,* more than one output satisfies the request) or because it is outside of the agent's understanding of the world (*e.g.,* a model trained in English will not understand a Spanish query, regardless of the information it contains). With the goal of minimizing error subject to constraints on human effort or human effort subject to constraints on error, the agent must determine whether to defer its decision and request information from the hazy oracle. If the agent chooses to defer its decision, it must additionally determine how best to integrate the additional, potentially noisy, information provided.

We show an abstraction of such an agent, modifying the terminology of previous work (Lemmer, Song, and Corso 2021), in Figure 2. The abstraction consists of three parts: the *task model* produces a prediction based on the input data, the *aggregation function* accepts one or more predictions corresponding to different human inputs from the task model and produces a combined prediction, and the *deferral function*—which consists of a *deferral score* and a threshold—determines whether or not a deferral should occur. While we discuss these as distinct entities, this is not a requirement: a recurrent neural network with a deferral score similar to SelectiveNet (Geifman and El-Yaniv 2019) would execute all three functions in a single step.

		Referring Expression Comprehension		
	VOT	Val	TestA	TestB
No Deferral	0.329 ± 6.6e^{-4}	8.82 ± 0.03	8.27 ± 0.04	9.67 ± 0.04
Perfect Deferral	0.276	2.07	1.92	2.82
Improvement	0.053	6.75	6.35	6.85
% Improvement	16.11%	76.53%	76.78%	70.84%

Table 1: Error for the task model with No Deferral (Err @ 0) and Perfect Deferral on two applications. A perfect deferral method can reduce error by over 76%. Err @ 0 is reported as mean and standard error of 100 trials. Full setup described under Exemplar Applications.

Figure 3: Whether a phrase is semantically ambiguous (left) or simply unclear to the model (right), a new human input can change an inference from incorrect (pink) to correct (blue). Best viewed in color.

Motivation We motivate the problem through the applications of single-target video object tracking, where the goal is to propagate bounding box drawn around an object in the first frame through all subsequent frames, and referring expression comprehension, where the goal is to draw a bounding box around the object described by a text query. We show the benefit of a perfect deferred inference method—that is, one that can select the best human input from the dataset—quantitatively in Table 1: for the validation split of the referring expression comprehension task, *using the best human input can reduce error by over 76%*.

In Figure 3 we demonstrate the benefit qualitatively by showing four human inputs and their matching outputs on the application of referring expression comprehension. On the left we see the more intuitive case, where the first expression, *yellow fire hydrant*, can be reasonably thought to refer to four objects, while *2nd hydrant* is mostly unambiguous.[1] On the right, we see a case where the referring expression *left banana* isn't truly ambiguous, but the model produces the wrong answer due to shortcomings in its understanding of language. Though the latter failure is more commonly considered a shortcoming of the model, the fact that a new referring expression can successfully solve both tasks demonstrates the potential benefit of deferred inference with modern, imperfect, models.

[1]While data collection was designed to produce unambiguous referring expressions, we find a non-negligible number of semantically ambiguous examples. This is likely due to the requirement that annotators make a guess for every expression.

Algorithm 1: Calculating DEV

```
DEV ← 0
DDC ← 1
while DDC ≤ 10 do
    tasks ← draw_tasks()
    DEV ← DEV + calc_error(tasks) / 10(len(tasks)+1)
    N ← 0
    while N < len(tasks) do
        cur_task ← find_task_to_defer(tasks, DDC)
        response ← get_new_input(cur_task)
        updated_task ← aggregate_fn(cur_task, response)
        update_tasks(tasks, updated_task)
        DEV ← DEV + calc_error(tasks) / 10(len(tasks)+1)
        N ← N + 1
    end while
    DDC ← DDC + 1
end while
```

Proposed Evaluation

Deferred Error Volume The exact performance of a deferred inference method is best measured as a combination of three different factors: i) the error, which is a property of the application; ii) the Deferral Rate (DR), which is the expected number of deferrals that will occur for each task; and, iii) the Deferral Depth Constraint (DDC), which is the maximum number of times that a task can be deferred. Since evaluating at only a single DR-DDC pair does not provide an adequate analysis of a deferral method, we propose the Deferred Error Volume (DEV), which finds the error at every potential combination of DR and DDC then calculates the volume under that surface.

While both DR and DDC are theoretically unconstrained, calculating the volume under a surface requires bounds to be placed. To produce these bounds, we make the least restrictive assumptions possible: the deferred inference method is capable of deferring every task at least once and the deferral function consists of a deferral score followed by a threshold. The former places an upper bound on DR at one and a lower bound on DDC at one, and the latter allows a thorough evaluation of the relationship between DR and error. We set an upper bound of ten on the DDC, which captures all practical deferral depths, and divide by ten to scale the width of this dimension to one. We discuss the implications of this upper bound in our results.

Since evaluation will be performed on finite datasets, the volume under the curve when using rectangular integration is the mean of error under all constraint sets:

$$\text{DEV} = \frac{1}{10(N+1)} \sum_{\text{DDC}=1}^{10} \sum_{n=0}^{N} \ell(DR = \frac{n}{N}, DDC) \ , \quad (1)$$

where $\ell(DR, DDC)$ is the error at a specific DR and DDC, and N is the number of tasks in the dataset. We show the calculation of DEV in Algorithm 1: after an initial error calculation with one randomly drawn human input for every task (draw_tasks), find_task_to_defer finds the highest deferral score where the DDC constraint is not exceeded, draws another human input from the dataset (get_new_input), uses the aggregation function to update the prediction, updates the DEV, and repeats the process.

Such a thorough dataset-based evaluation has only one major requirement: *there must be a method by which deferral responses can be provided*. This requirement can be satisfied in a number of ways: the deferral response may be of the same form as the initial piece of human information with a dataset containing multiple pieces of human input per task (the approach of this work), the deferral query and response may be from a set of pre-defined attributes (Branson et al. 2010), or an external agent capable of answering additional queries—potentially with access to oracle knowledge—may be developed (Uehara, Duan, and Harada 2022).

Marginals To illustrate the effect of individual constraints on a method's performance, we marginalize out the DR and DDC constraints and plot the result, referring to the measurement as *mean error*. Notably, calculating these marginals requires no inferences beyond those already used to calculate the DEV.

Error To provide an additional intuitive measure of the performance improvement enabled by a deferral method, we report error at two specific locations: deferral rate of zero (Err @ 0), which is the base error of the task model, and deferral rate of one (Err @ 1), which corresponds to the error when the number of deferrals is equal to the number of tasks (if DDC > 1, this does not mean every task will have exactly one deferral). Since these errors correspond to the first and last points on the relevant marginal plot, no additional calculation is required to obtain them.

Proposed Method

We propose a straightforward method for deferred inference that can easily be applied to many state-of-the-art task models. Underlying our method are two assumptions: first, we assume the task model output is a distribution. This is trivial for a softmax output but may require additional consideration for other output formats (Harakeh, Smart, and Waslander 2020). Second, we assume the deferral function is based on this distribution. With these assumptions, we can then use a belief update as the aggregation function. If we treat human inputs $h_1, ..., h_n$ as independent and represent the non-deferrable portion of the input (*e.g.,* an image) as x, the probability of a specific output, y, is:

$$p(y|x, h_1, ..., h_n) \propto \prod_{n=1}^{N} p(y|x, h_n) \ . \quad (2)$$

Though this formulation permits the deferral and its response to take many forms, we treat the deferral as a request for the human operator to rephrase the initial input. This allows our formulation to be rapidly applied to new research in the machine learning space, as it does not require novel datasets or architectural changes.

Implementation of our formulation is illustrated intuitively for a classification task such as Referring Expression Comprehension (Figure 4): an initial image, x, and referring expression, h_1, are given to the task model, resulting in a

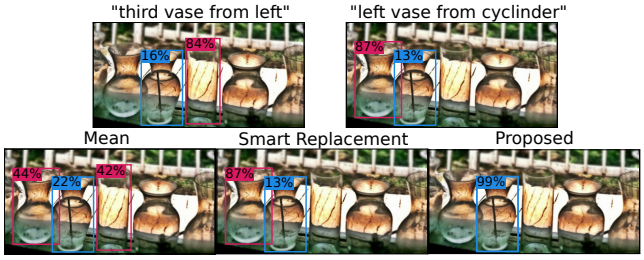

Figure 4: Our proposed aggregation function quickly combines complementary information to achieve higher confidence and accuracy than previous methods such as taking the mean of two outputs (Hatori et al. 2018) or selecting the better output (Smart Replacement (Lemmer, Song, and Corso 2021)). Target object boxed in blue. Original image cropped vertically for space. View in color.

softmax output. If the deferral function decides that inference should be deferred based on this output, h_2 is solicited and passed through the task model alongside x, resulting in a second softmax output. The two output vectors are then multiplied elementwise, and the resulting vector is normalized. The deferral function is then executed on this normalized vector to determine if another deferral is necessary. As we see, combining the inferences in this way demonstrates a major benefit of our method: it quickly identifies the target object with high certainty, while aggregation functions such as using the better of the two outputs (Lemmer, Song, and Corso 2021) or taking the mean of the two outputs (Hatori et al. 2018) would perform additional deferrals or return an incorrect answer depending on the given constraints.

Exemplar Applications

Single-Target VOT

Goal In single-target Video Object Tracking (VOT), a human defines the task by drawing a bounding box around an object in the first frame of a video. Using only previous frames (the online tracking setting) the model then propagates this bounding box through the video. Despite the relatively low dimension of the input space, deferred inference is properly motivated for this application: driven by the sensitivity of tracking algorithms to their initializations, perturbation tests are a standard component of evaluation (Wu, Lim, and Yang 2013; Kristan et al. 2016).

Model and Dataset Since it is the only VOT dataset, to our knowledge, that contains multiple annotations per tracked object, we perform our analysis using the crowdsourced data from Lemmer *et al.* (Lemmer, Song, and Corso 2021). This dataset consists of nine first-frame annotations for every video in the OTB-100 dataset (Wu, Lim, and Yang 2013). To differentiate between low-quality inferences and inferences performed on an incorrectly selected object, we remove instances in this dataset where the initialization has an IoU of less than 0.5 with the gold-standard initialization. As our task model, we use the ToMP tracker (Mayer et al. 2022) with a ResNet-50 backbone (He et al. 2016) and weights provided by the original authors.

Base Error Metric We measure performance using the mean intersection-over-union (IoU), which is commonly used in the evaluation of the VOT application (Wu, Lim, and Yang 2013; Kristan et al. 2016). To maintain the notion of error, we subtract the IoU from its maximum value of one. Unlike previous evaluations, we allow the model's inference to continue when the object track is lost: resetting the tracker requires a ground-truth box on every frame, which is intractable both in a real-world application and in the context of our implementation of deferred inference.

Deferral Implementation The output of ToMP is a single bounding box at every frame. To convert this to a distribution for our deferral and aggregation functions, we first produce stochastic bounding boxes by performing each inference 100 times with Monte Carlo dropout enabled (Gal and Ghahramani 2016) similar to previous work on object detection (Harakeh, Smart, and Waslander 2020). Every bounding box is represented as a tuple of (TLX, TLY, Width, Height) and expectation maximization (Dempster, Laird, and Rubin 1977) is performed on every frame to transform these representations into a Mixture of Gaussians. To determine the number of Gaussians for expectation maximization, we use DBSCAN (Ester, Kriegel, and Xu 1996) with epsilon 10 and minimum samples 20, which provided the lowest deferral-free error among the parameter set $\epsilon \in \{1, 3, 5, 10, 15, 20, 50\}$ and min_samples $\in \{3, 5, 10, 15, 20\}$. We use Scikit-Learn (Pedregosa et al. 2011) for both Expectation Maximization and DBSCAN.

Using these distributions, the deferral score is produced by randomly sampling 500 bounding box pairs and measuring the mean IoU between them. For both our method and baselines, we create an output bounding box by taking 10,000 samples from the mixture and using the one with the highest likelihood. For our method, these samples are scattered by adding a normally-distributed random value with standard deviation 7 to every dimension, which allows us to combine distributions that are close in Euclidean space but several standard deviations apart.

Referring Expression Comprehension

Goal In referring expression comprehension, a task is defined by an image and text query, and the task model draws a bounding box around the object described by the text. The high dimensionality of the input space means that there is much room for both semantic ambiguity and gaps in the task model's knowledge that can be corrected or clarified after a deferral.

Model and Dataset For the task model, our evaluation uses the UNITER architecture (Chen et al. 2020), which formulates referring expression comprehension as classification over a set of externally-provided bounding boxes. We provide ground-truth detections for these bounding boxes to minimize the influence of an external object detector. We train and evaluate on the RefCOCO (Kazemzadeh et al. 2014) dataset because it contains multiple references to all but one target object, which is substantially better than both the RefCOCO+ (Kazemzadeh et al. 2014), and RefCOCOg (Mao et al. 2016) datasets. Our model is trained

	VOT		RefExp (Val)		RefExp (TestA)		RefExp (TestB)	
Method	DEV	Err @ 1	DEV	Err @ 1	DEV	Err @ 1	DEV	Err @ 1
Naive R.	$0.3279 \pm 1.6e^{-4}$	$0.3269 \pm 5.7e^{-4}$	$6.92 \pm 7.9e^{-3}$	$6.56 \pm 7.6e^{-3}$	$5.92 \pm 8.2e^{-3}$	$5.65 \pm 1.1e^{-2}$	$7.60 \pm 9.1e^{-3}$	$6.90 \pm 9.3e^{-3}$
Mean	$0.3286 \pm 1.4e^{-4}$	$0.3277 \pm 4.4e^{-4}$	$7.26 \pm 8.2e^{-3}$	$6.46 \pm 7.5e^{-3}$	$6.48 \pm 1.1e^{-2}$	$5.68 \pm 1.0e^{-2}$	$8.09 \pm 1.1e^{-2}$	$7.20 \pm 1.1e^{-2}$
Consensus	N/A	N/A	$7.94 \pm 7.9e^{-3}$	$7.47 \pm 8.1e^{-3}$	$7.33 \pm 1.2e^{-2}$	$6.92 \pm 1.2e^{-2}$	$8.80 \pm 1.2e^{-2}$	$8.32 \pm 1.1e^{-2}$
Smart R.	$0.3280 \pm 1.6e^{-4}$	$0.3264 \pm 5.0e^{-4}$	$6.51 \pm 5.6e^{-3}$	$5.68 \pm 5.7e^{-3}$	$5.41 \pm 8.1e^{-3}$	$4.55 \pm 7.5e^{-3}$	$7.29 \pm 1.0e^{-2}$	$6.17 \pm 9.3e^{-3}$
Ours	**$0.3263 \pm 1.3e^{-4}$**	**$0.3245 \pm 4.4e^{-4}$**	**$6.13 \pm 6.4e^{-3}$**	**$5.23 \pm 5.9e^{-3}$**	**$5.16 \pm 8.6e^{-3}$**	**$4.24 \pm 7.8e^{-3}$**	**$6.92 \pm 8.6e^{-3}$**	**$5.74 \pm 8.7e^{-3}$**

Table 2: The DEV and Err @ 1 metrics for baselines and our method (Err @ 0 shown in Table 1). Our method performs best across all applications and splits by a significant margin.

on a single GeForce GTX Titan XP GPU using the training settings given by the original authors with a few small modifications: we use full precision floating point operations, adjust the batch size from 128 to 64, and accumulate gradients over two steps.

Base Error Metric Consistent with previous work (Mao et al. 2016), performance on this application is measured using an indicator function. Error is zero if the predicted bounding box has an IoU of greater than 0.5 with the ground-truth bounding box, and 100 otherwise. Aggregate error can then be interpreted as the percentage of tasks that are completed incorrectly. We maintain the val, testA, and testB splits from previous works (Yu et al. 2016), but note our evaluation measures per-task performance instead of per-phrase performance, making it incorrect to directly compare our results to other evaluations.

Deferral Implementation Because the UNITER referring expression comprehension model outputs a softmax distribution, we use entropy as our deferral score and implement our proposed aggregation function by multiplying and normalizing the outputs. We improve the output's ability to detect ambiguity by performing Monte Carlo dropout with 100 passes, matching the number of passes in the original work (Gal and Ghahramani 2016). We discuss the importance of using MC dropout in our technical appendix.

Experiments

Baselines We compare our proposed method to four aggregation functions adapted from previous work:

- Naive Replacement: If a deferral is performed, the most recent input is always used. If no DDC is specified this is equivalent to a selective prediction approach (Chow 1970; Geifman and El-Yaniv 2017), where the user must restart the task if the inference is declined.
- Mean: If the inference is deferred, the mean of DDC new inputs is used. This is equivalent to the aggregation function of Hatori *et al.* (Hatori et al. 2018), who implicitly defined a DDC of one.
- Consensus: If a deferral is performed, DDC new inputs are taken and the consensus of all outputs is returned as the answer. If there is no consensus, an answer is chosen randomly from the potential outputs with equal occurrences. This is a basic approach often used in crowdsourcing (Deng et al. 2009; Song et al. 2019). We do not implement this baseline on the VOT application due to the high number of potential outputs.
- Smart Replacement: If inference is deferred, the deferral score between all responses is compared, and the output corresponding to the best deferral score is used. As with the Mean baseline, we extend the original work (Lemmer, Song, and Corso 2021) by allowing the DDC to be greater than one.

Results We see in Table 2 that deferred inference improves over the no deferral condition both on the mean (DEV) and at a deferral rate of one (err @ 1) for all methods and problem settings. In other words, any of the evaluated aggregation functions are better than the no deferral condition. Further, our proposed aggregation function outperforms all baselines in all settings on the evaluated metrics, and reduces error between the deferral-free condition (Table 1-No Deferral) and deferral rate 1 (Table 2-Err @ 1): error decreases by 1.37% for VOT, 40.7% for RefExp-Val, 48.7% for RefExp-TestA, and 40.6% for RefExp-TestB. In other words, *our method is effective on two very different applications, and can reduce error by over 48% (from 8.27% to 4.24%) without any change to the model.*

Marginals We now consider the effect of individual constraints by marginalizing out the DR (Figure 5-Left) and DDC (Figure 5-Right). By examining the former, we aim to answer two specific questions: what is the effect of our DDC range on the ordinal results of the DEV metric, and what is the effect of the DDC on performance? For the former question, we see that our method is unambiguously better—that is, best or within one standard error of best at all DDC—on both tasks. However, the improved performance of our method on the VOT task is primarily due to its ability to effectively handle greater DDCs: if our evaluation were limited to DDC ≤ 2, the DEV would be within one standard error of the Smart Replacement and Mean baselines.

Further consideration of the interaction between DDC and mean error provides meaningful insight into both our method and the findings of previous work. First, while other methods begin to meaningfully degrade as DDC increases, our method does not show such severe trends. Next, the DDC of one used in previous work (Lemmer, Song, and Corso 2021; Uehara, Duan, and Harada 2022) has meaningful shortcomings: all aggregation functions, with the exception of Mean, are improved by increasing the DDC beyond one on the referring expression comprehension task and, on the video object tracking task, the finding of previ-

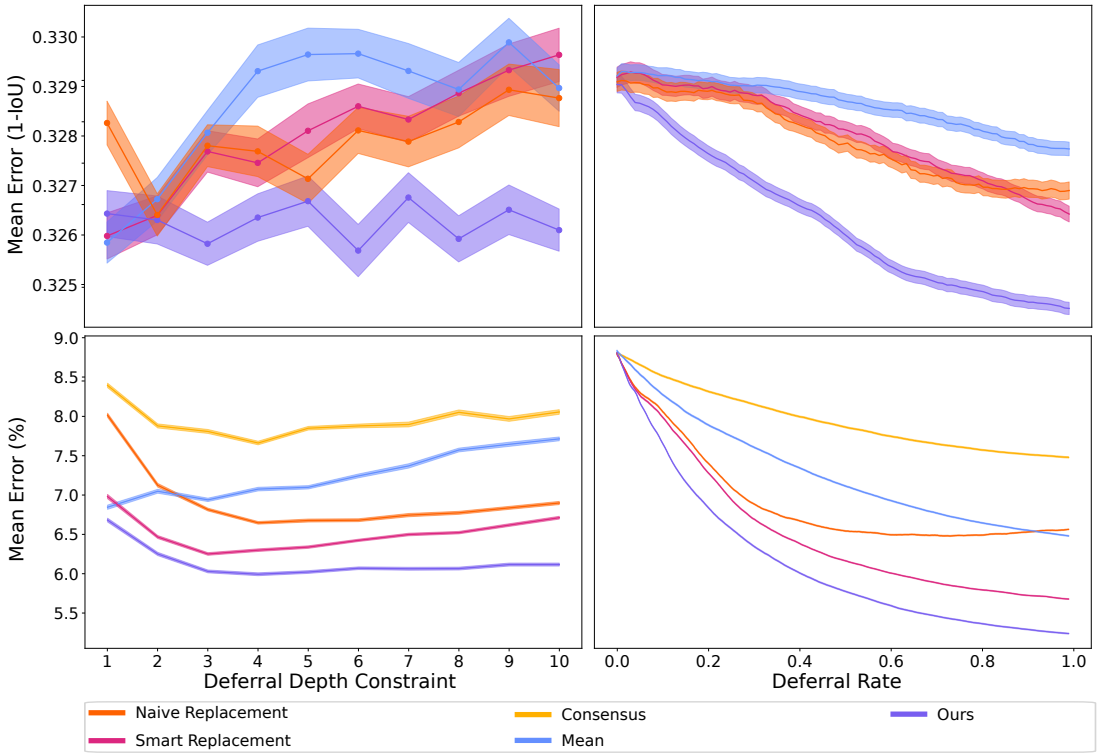

Figure 5: Marginal plots showing the effect of the DDC (left) and DR (right) on the VOT (top) and Referring Expression Comprehension (val split) (bottom) applications. Shaded area represents one standard error across 100 trials. View in color.

ous work (Lemmer, Song, and Corso 2021) that Smart Replacement is significantly better than Naive Replacement is only supported at the DDC of one used in their evaluation.

When the DDC is marginalized, our method is best or within one standard error of best at all DRs. Broadly speaking, the behavior of this marginal is as expected: error decreases as DR increases for all aggregation functions with the exception of naive replacement, which increases at higher DRs. As noted when naive replacement was first used as a baseline (Lemmer, Song, and Corso 2021), this is due to the tendency to defer correct inferences at higher DRs and replace them with potentially low-quality human inputs.

Discussion

Alternate Task Definitions While our work is motivated by scenarios where human-provided information is used to define the task, there are two other hazy oracle formulations that should be considered: for applications such as Keypoint-Conditioned Viewpoint Estimation (Szeto and Corso 2017) where the task could be performed without human information, the model could choose to solicit the hazy oracle only when a human-provided keypoint could change the answer—which is often not the case (Lemmer and Corso 2021). The task could also be defined by an automated hazy oracle: for example, Gurari *et al.* show that different segmentation algorithms work better in different cases (Gurari et al. 2016), which would have implications for a subsequent Video Object Segmentation (Perazzi et al. 2016). Our evaluation and solution extend trivially to both cases.

Deployment When shifting from dataset-based evaluation to practical human interaction, a few additional factors must be considered. First, we must define a threshold on the deferral score that targets an error or deferral rate. This requires an additional pass on a validation set and may use a method analogous to Selective Guaranteed Risk (Geifman and El-Yaniv 2017) if performance guarantees are required. Second, the datasets used for the experiments in this work were crowdsourced meaning that, although our findings are directly applicable to a crowdsourcing setting, factors that are dependent on an individual human are not evaluated: individuals have different biases and variances and the quality of the provided inputs may be dependent on the number of deferrals that have occurred. Works motivated by interaction with individuals should consider this during dataset procurement and evaluation.

Conclusion

Despite the intuitive benefit of deferred inference when information is provided by hazy oracles, previous works have performed only surface-level analyses under limited experiments, leading to an inability to effectively develop and compare methods. Through formalization of deferred inference, a novel evaluation metric, and demonstration of a straightforward method that provides meaningful improvement across two disparate applications, we hope to enable and motivate further research into this impactful problem.

Acknowledgments

Toyota Research Institute ("TRI") provided funds to assist the authors with their research but this article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity. This work was also partially supported by NSF CNS 1628987, the Google Faculty Research Award, and a gift from NEC.

References

Agustsson, E.; Uijlings, J. R. R.; and Ferrari, V. 2019. Interactive Full Image Segmentation by Considering All Regions Jointly. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11622–11631. Long Beach, CA, USA: IEEE Press.

Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sunderhauf, N.; Reid, I.; Gould, S.; and van den Hengel, A. 2018. Vision-and-Language Navigation: Interpreting Visually-Grounded Navigation Instructions in Real Environments. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3674–3683. Salt Lake City, UT, USA: IEEE Press.

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Zitnick, C. L.; and Parikh, D. 2015. VQA: Visual Question Answering. In *Proceedings of the 2015 IEEE International Conference on Computer Vision*, 2425–2433. Santiago, Chile: IEEE Press.

Bhattacharya, N.; Li, Q.; and Gurari, D. 2019. Why Does a Visual Question Have Different Answers? In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision*, 4270–4279. Seoul, South Korea: IEEE Press.

Bondi, E.; Koster, R.; Sheahan, H.; Chadwick, M.; Bachrach, Y.; Cemgil, T.; Paquet, U.; and Dvijotham, K. 2022. Role of Human-AI Interaction in Selective Prediction. In *Proceedings of the 2022 AAAI Conference on Artificial Intelligence*, volume 36, 5286–5294. Virtual: AAAI Press.

Branson, S.; Wah, C.; Schroff, F.; Babenko, B.; Welinder, P.; Perona, P.; and Belongie, S. 2010. Visual Recognition with Humans in the Loop. In *Proceedings of the 2010 European Conference on Computer Vision*, volume 6314, 438–451. Berlin, Heidelberg: Springer Berlin Heidelberg.

Chen, Y.-C.; Li, L.; Yu, L.; Kholy, A. E.; Ahmed, F.; Gan, Z.; Cheng, Y.; and Liu, J. 2020. UNITER: UNiversal Image-TExt Representation Learning. In *Proceedings of the 2020 European Conference on Computer Vision*, 104–120. Virtual: Springer.

Choi, M.; Park, C.; Yang, S.; Kim, Y.; Choo, J.; and Hong, S. R. 2019. AILA: Attentive Interactive Labeling Assistant for Document Classification through Attention-Based Deep Neural Networks. In *Proceedings of the 2019 Conference on Human Factors in Computing Systems*, 1–12. Glasgow, Scotland, UK: ACM Press.

Chow, C. 1970. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1): 41–46.

Cortes, C.; DeSalvo, G.; and Mohri, M. 2016. Boosting with Abstention. In *Proceedings of the 2016 Conference on Advances in Neural Information Processing Systems*, 9. Barcelona, Spain: Curran Associates.

Das, A.; Kottur, S.; Gupta, K.; Singh, A.; Yadav, D.; Moura, J. M. F.; Parikh, D.; and Batra, D. 2017. Visual Dialog. In *Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 326–335. Honolulu, HI: IEEE Press.

Dempster, A. P.; Laird, N. M.; and Rubin, D. B. 1977. Maximum Likelihood from Incomplete Data Via the *EM* Algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1): 1–22.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. Miami, FL: IEEE Press.

Ester, M.; Kriegel, H.-P.; and Xu, X. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 226–231. Portland, OR, USA: AAAI Press.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proceedings of the 2016 International Conference on Machine Learning*, 1050–1059. New York, New York, USA: PMLR.

Geifman, Y.; and El-Yaniv, R. 2017. Selective Classification for Deep Neural Networks. In *Proceedings of the 2017 conference on Advances in Neural Information Processing Systems*, 4878–4887. Long Beach, CA, USA: Curran Associates.

Geifman, Y.; and El-Yaniv, R. 2019. SelectiveNet: A Deep Neural Network with an Integrated Reject Option. In *Proceedings of the 36th international conference on Machine learning*, 2151–2159. Long Beach, CA, USA: ACM Press.

Geifman, Y.; Uziel, G.; and El-Yaniv, R. 2019. Bias-Reduced Uncertainty Estimation for Deep Neural Classifiers. In *Proceedings of the Seventh International Conference on Learning Representations*, 1–14. New Orleans, LA.

Gordon, M. L.; Zhou, K.; Patel, K.; Hashimoto, T.; and Bernstein, M. S. 2021. The Disagreement Deconvolution: Bringing Machine Learning Performance Metrics In Line With Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 1–14. Yokohama Japan: ACM.

Gouravajhala, S. R.; Yim, J.; Desingh, K.; Huang, Y.; Jenkins, O. C.; and Lasecki, W. S. 2018. EURECA: Enhanced Understanding of Real Environments via Crowd Assistance. In *Proceedings of the Sixth AAAI Conference on Human Computation and Crowdsourcing*, 31–40. Zurich, Switzerland: AAAI Press.

Gurari, D.; Jain, S. D.; Betke, M.; and Grauman, K. 2016. Pull the Plug? Predicting If Computers or Humans Should Segment Images. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, 382–391. Las Vegas, NV, USA: IEEE Press.

Gurari, D.; Li, Q.; Stangl, A. J.; Guo, A.; Lin, C.; Grauman, K.; Luo, J.; and Bigham, J. P. 2018. VizWiz Grand Challenge: Answering Visual Questions from Blind People. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3608–3617. Salt Lake City, UT, USA: IEEE Press.

Harakeh, A.; Smart, M.; and Waslander, S. L. 2020. BayesOD: A Bayesian Approach for Uncertainty Estimation in Deep Object Detectors. In *Proceedings of the 2020 IEEE International Conference on Robotics and Automation*, 87–93. Virtual: IEEE Press.

Hatori, J.; Kikuchi, Y.; Kobayashi, S.; Takahashi, K.; Tsuboi, Y.; Unno, Y.; Ko, W.; and Tan, J. 2018. Interactively Picking Real-World Objects with Unconstrained Spoken Language Instructions. In *Proceedings of the 2018 IEEEE International Conference on Robotics and Automation*, 3774–3781. Brisbane, Australia: IEEE Press.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 770–778. Las Vegas, NV: IEEE Press.

Ipeirotis, P. G.; Provost, F.; and Wang, J. 2010. Quality management on Amazon Mechanical Turk. In *Proceedings of the 2010*

ACM SIGKDD Workshop on Human Computation, 64–67. Washington DC: ACM Press.

Jain, S. D.; and Grauman, K. 2016. Click Carving: Segmenting Objects in Video with Point Clicks. In *Proceedings of the Fourth AAAI Conference on Human Computation and Crowdsourcing*, 89–98. Austin, Texas, USA: AAAI Press.

Kazemzadeh, S.; Ordonez, V.; Matten, M.; and Berg, T. 2014. ReferItGame: Referring to Objects in Photographs of Natural Scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 787–798. Doha, Qatar: Association for Computational Linguistics.

Kristan, M.; Matas, J.; Leonardis, A.; Vojíř, T.; Pflugfelder, R.; Fernández, G.; Nebehay, G.; Porikli, F.; and Čehovin, L. 2016. A Novel Performance Evaluation Methodology for Single-Target Trackers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11): 2137–2155.

Lemmer, S. J.; and Corso, J. J. 2021. Ground-Truth or DAER: Selective Re-Query of Secondary Information. In *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision*, 703–714. Virtual: IEEE Press.

Lemmer, S. J.; Song, J. Y.; and Corso, J. J. 2021. Crowdsourcing More Effective Initializations for Single-Target Trackers Through Automatic Re-querying. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 1–13. Yokohama Japan: ACM Press.

Mahendru, A.; Prabhu, V.; Mohapatra, A.; Batra, D.; and Lee, S. 2017. The Promise of Premise: Harnessing Question Premises in Visual Question Answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 926–935. Copenhagen, Denmark: Association for Computational Linguistics.

Mao, J.; Huang, J.; Toshev, A.; Camburu, O.; Yuille, A.; and Murphy, K. 2016. Generation and Comprehension of Unambiguous Object Descriptions. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 11–20. Las Vegas, NV, USA: IEEE Press.

Mayer, C.; Danelljan, M.; Bhat, G.; Paul, M.; Paudel, D. P.; Yu, F.; and Van Gool, L. 2022. Transforming Model Prediction for Tracking. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8731–8740. New Orleans, LA, USA: IEEE Press.

Mees, O.; and Burgard, W. 2020. Composing Pick-and-Place Tasks By Grounding Language. In *Proceedings of the 2020 International Symposium on Experimental Robotics*, 491–501. La Valletta, Malta: Springer.

Mozannar, H.; and Sontag, D. 2020. Consistent Estimators for Learning to Defer to an Expert. In *Proceedings of the 2020 International Conference on Machine Learning*, 7076–7087. Virtual: PMLR.

Nyga, D.; Roy, S.; Paul, R.; Park, D.; Pomarlan, M.; Beetz, M.; and Roy, N. 2018. Grounding Robot Plans from Natural Language Instructions with Incomplete World Knowledge. In *Proceedings of the 2018 Conference on Robot Learning*, 714–723. Zurich, Switzerland: PMLR.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830.

Perazzi, F.; Pont-Tuset, J.; McWilliams, B.; Van Gool, L.; Gross, M.; and Sorkine-Hornung, A. 2016. A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, 724–732. Las Vegas, NV: IEEE Press.

Russakovsky, O.; Li, L.-J.; and Li, F.-F. 2015. Best of both worlds: Human-machine collaboration for object annotation. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*, 2121–2131. Boston, MA, USA: IEEE Press.

Sharma, P.; Sundaralingam, B.; Blukis, V.; Paxton, C.; Hermans, T.; Torralba, A.; Andreas, J.; and Fox, D. 2022. Correcting Robot Plans with Natural Language Feedback. In *Proceedings of the 2022 Conference on Robotics: Science and Systems*, 1–12. New York, New York, USA: MIT Press.

Shridhar, M.; and Hsu, D. 2018. Interactive Visual Grounding of Referring Expressions for Human-Robot Interaction. In *Proceedings of Robotics: Science and Systems 2018*, 1–9. Pittsburgh, Pennsylvania, United States: MIT Press.

Song, J. Y.; Lemmer, S. J.; Liu, M. X.; Yan, S.; Kim, J.; Corso, J. J.; and Lasecki, W. S. 2019. Popup: reconstructing 3D video using particle filtering to aggregate crowd responses. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, 558–569. Marina del Ray, California: ACM Press.

Szeto, R.; and Corso, J. J. 2017. Click Here: Human-Localized Keypoints as Guidance for Viewpoint Estimation. In *Proceedings of the 2017 IEEE/CVF International Conference on Computer Vision*, 1604–1613. Venice: IEEE Press.

Uehara, K.; Duan, N.; and Harada, T. 2022. Learning To Ask Informative Sub-Questions for Visual Question Answering. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 4681–4690. New Orleans, LA, USA: IEEE Press.

Uijlings, J.; Konyushkova, K.; Lampert, C. H.; and Ferrari, V. 2018. Learning Intelligent Dialogs for Bounding Box Annotation. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9175–9184. Salt Lake City, UT: IEEE Press.

Uijlings, J. R. R.; Andriluka, M.; and Ferrari, V. 2020. Panoptic Image Annotation with a Collaborative Assistant. In *Proceeedings of the 28th ACM International Conference on Multimedia*, 3302–3310. Virtual: ACM Press.

Wu, Y.; Lim, J.; and Yang, M.-H. 2013. Online Object Tracking: A Benchmark. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2411–2418. Portland, OR, USA: IEEE Press.

Yildirim, M. Y.; Ozer, M.; and Davulcu, H. 2019. Leveraging Uncertainty in Deep Learning for Selective Classification. *arXiv:1905.09509 [cs, math, stat]*.

Yu, L.; Poirson, P.; Yang, S.; Berg, A. C.; and Berg, T. L. 2016. Modeling Context in Referring Expressions. In *Proceedings of the 2016 European Conference on Computer Vision*. Amsterdam, The Netherlands: Springer.