Dialog Policy Learning for Joint Clarification and Active Learning Queries

Aishwarya Padmakumar, [1] **Raymond J. Mooney** [2]
[1] Amazon Alexa AI *
[2] University of Texas at Austin
padmakua@amazon.com, mooney@cs.utexas.edu

Abstract

Intelligent systems need to be able to recover from mistakes, resolve uncertainty, and adapt to novel concepts not seen during training. Dialog interaction can enable this by the use of clarifications for correction and resolving uncertainty, and active learning queries to learn new concepts encountered during operation. Prior work on dialog systems has either focused on exclusively learning how to perform clarification/ information seeking, or to perform active learning. In this work, we train a hierarchical dialog policy to jointly perform *both* clarification and active learning in the context of an interactive language-based image retrieval task motivated by an online shopping application, and demonstrate that jointly learning dialog policies for clarification and active learning is more effective than the use of static dialog policies for one or both of these functions.

1 Introduction

The ability to understand and communicate in natural language can improve the accessibility of systems such as robots, home devices and computers to non-expert users. Since language is often be ambiguous, it is desirable for such systems to engage in a dialog with the user to clarify their intentions and obtain missing information. We use *clarification* to refer to any dialog act that enables the system to better understand an ongoing user request. Common clarification questions obtain or clarify the value of a slot or argument that is part of a goal the user is trying to communicate.

A particular application may also contain domain-specific vocabulary or concepts that were not encountered during training. For example, a system in a shopping domain may need to be updated with the introduction of new clothing styles. Hence, it is desirable for a system to adapt to the operating environment using information from user interactions. We use the term *active learning* to refer to dialog acts used to obtain such knowledge with the primary purpose of improving the underlying language understanding model and thereby improving performance on future interactions.

Prior work on dialog and user interaction typically focuses either exclusively on clarification (Young et al. 2013; Padmakumar, Thomason, and Mooney 2017), or active learning (Woodward and Finn 2017; Padmakumar, Stone, and Mooney 2018). The primary contributions of this work are introducing a dialog task that combines both clarification and active learning, and learning a corresponding dialog policy for this setting that outperforms a static baseline policy. Specifically, we train a hierarchical dialog policy to jointly learn to choose clarification and active learning queries in interactive image retrieval for a fashion domain.

Speaker	Action Type	Simulated Dialog Act	Natural Language Gloss
Agent	-	-	Is there something I can help you find?
User	Target Description	Mixed Pain Signals Spaghetti Top	I'm looking for a mixed pain signals spaghetti top
Agent	Clarification	Clarification: Pink	Would you like one which is pink?
User	Binary Label	1	Yes
Agent	Label Query (Opportunistic)	Label Query: Sleeveless 	Would you describe this item as sleeveless?
User	Binary Label	1	Yes
Agent	Example Query (On-topic)	Example Query: Spaghetti	Can you share an image of an item with spaghetti straps?
User	Image		Here is an example.
Agent	Guess		Is this what you were searching for?
User	Success: 0/1	1	Yes

Figure 1: A stylized sample interaction. In this work, we use simulated dialog acts but the natural language glosses represent how such a dialog could look in an end application.

A sample interaction is shown in Figure 1. We consider an application where a dialog system is combined with a retrieval system to help a customer find an article of clothing. Instead of just showing a large number of retrieved results, the dialog system attempts to use clarifications to refine the search query, and active learning questions to obtain labelled examples for novel concepts unseen during training.

Task-oriented dialog often requires the system to identify one or more user goals using a slot-filling model (Young et al. 2013). These systems learn to choose between a set of clarification questions that confirm or acquire the value of various slots. However, for tasks such as natural language image retrieval, it is difficult to extend the slot-filling

*This work was done as a part of the first author's PhD at the University of Texas at Austin

paradigm for clarification, as there is no standard set of slots into which descriptions of images can be divided. Also, learned models are needed to identify aspects such as objects or attributes, which are difficult to pre-enumerate.

Some tasks such as GuessWhat?! (De Vries et al. 2017) or discriminative question generation (Li et al. 2017) allow the system to ask unconstrained natural language clarification questions. However they require specially designed models to ensure that learned questions actually decrease the search space (Lee, Heo, and Zhang 2018; Zhang et al. 2018). Such open ended questions are also difficult to answer in simulation, which is often necessary for learning good dialog policies. Hence, in these tasks, the system often learns to ask "easy" questions that can be reliably answered by a learned answering module (Zhu, Zhang, and Metaxas 2017).

In this work, we explore a middle-ground approach with a form of attribute-based clarification (Farhadi et al. 2009). We use the term "attribute" to refer to a mix of concepts including categories such as "shirt" or "dress", more conventional attributes such as colors, and domain specific attributes such as "sleeveless" and "V-neck". Although we work with a dataset that contains a fixed set of attributes annotated for each image, we simulate the setting where novel visual attributes are encountered at test time.

Dialog interaction can also be used to improve an underlying model using Opportunistic Active Learning (OAL) (Padmakumar, Stone, and Mooney 2018). Active learning allows a system to identify unlabeled examples which, if labeled, are most likely to improve the underlying model. OAL (Thomason et al. 2017) incorporates such queries into an interactive task in which an agent may ask users questions that are irrelevant to the current dialog interaction to improve performance in future dialog interactions. Opportunistic queries are more expensive than traditional active learning queries as they may distract from the task at hand, but they can allow the system to perform more effective lifelong learning. Such queries have been shown to improve performance in interactive object retrieval (Padmakumar, Stone, and Mooney 2018). However, this, and other works in reinforcement learning (RL) of policies for active learning (Fang, Li, and Cohn 2017) do not account for the presence of other interactive actions such as clarification.

We present a dialog task that combines natural language image retrieval with *both* OAL *and* attribute-based clarification. We then learn a hierarchical dialog policy that jointly learns to choose both appropriate clarification and active learning questions in a setting containing both uncertain visual classifiers and novel concepts not seen during training. We observe that in our challenging setup, it is necessary to jointly learn dialog policies for choosing clarification and active learning questions to improve performance over employing one-shot retrieval with no interaction.

2 Related Work

Slot-filling style clarifications (Young et al. 2013) have been shown to be useful for a variety of domains including restaurant recommendation (Williams, Raux, and Henderson 2016), restaurant reservation (Bordes, Boureau, and Weston 2017), movie recommendation and question answering (Dodge et al. 2016), issuing commands to robots (Deits et al. 2013; Thomason et al. 2015) and converting natural language instructions to code (Chaurasia and Mooney 2017). Other tasks such as GuessWhat?! (De Vries et al. 2017), playing 20 questions (Hu et al. 2018), relative captioning (Guo et al. 2018) and discriminative question generation (Li et al. 2017) enable very open-ended clarification. Some works bring the setup closer to human-human conversations by allowing speakers to interrupt each other (Manuvinakurike, DeVault, and Georgila 2017). In this work, we take an intermediate approach that allows finer-grained clarification than slot filling, but constrained so that reasonably accurate answers can be provided in simulation.

Most of the above works learn dialog policies for clarification using RL (Padmakumar, Thomason, and Mooney 2017; Wen et al. 2016; Strub et al. 2017; Hu et al. 2018). Some use information from clarifications to improve the underlying language-understanding model (Thomason et al. 2015; Padmakumar, Thomason, and Mooney 2017). Such improvement is implicit in end-to-end dialog systems (Wen et al. 2016; De Vries et al. 2017; Hu et al. 2018). Instead, we use explicit active learning to improve the underlying perceptual model used for language grounding. Some previous work uses visual attributes for clarification (Dindo and Zambuto 2010; Parde et al. 2015), but they do not use this information to improve the underlying language understanding model. There is also prior work on using active learning to elicit better user feedback to learn a better reward function to optimize dialog policies for task oriented dialog (Su et al. 2016). This direction is complementary to our work which is aimed at using active learning to improve the underlying language understanding model.

Hierarchical dialog policies have been designed for multi-task dialog systems where a top level policy alternates between subtasks such as booking a hotel or a flight, and a lower level policy that selects primitive dialog actions to complete the subtasks (Peng et al. 2017; Budzianowski et al. 2017). Some hierarchical techniques such as feudal RL can enable dialog systems to scale to handle domains with a large number of slots (Casanueva et al. 2018). More related is (Zhang, Zhao, and Yu 2018), who design a hierarchical policy for visual dialog that has a low level policy for choosing clarification questions, and a high level decision policy to choose between clarification and guessing. Our work can be considered an extension of this framework that additionally accounts for active learning by including an additional low level policy for choosing active learning queries, and expanding the top level decision policy to choose between clarification, active learning as well as guessing.

Active learning has traditionally used hand-coded sample-selection metrics, such as uncertainty sampling (Settles 2010). Recent work on active learning in dialog/RL setups include using slot-filling style active learning questions to learn new names for known perceptual concepts (Yu, Eshghi, and Lemon 2017), sequentially identify exampled to be labeled for a static task (Fang, Li, and Cohn 2017), deciding between predicting a label for a specific example or requesting for it to be labelled (Woodward and Finn 2017),

and jointly learning a data selection heuristic, data representation, and prediction function (Bachman, Sordoni, and Trischler 2017). However, most of these (except (Woodward and Finn 2017)) do not involve a trade-off between active learning and task completion. None of them incorporate clarification questions.

Most similar to our work is (Padmakumar, Thomason, and Mooney 2017) which concerns learning a policy to trade-off opportunistic active learning questions to improve classifiers, and using these to ground natural-language descriptions of objects. However, instead of assuming a cold-start condition where the system cannot initially ground any descriptions before asking queries, we consider a warm-start condition closer to most real-world scenarios. We use a pretrained classifier and expect active learning to primarily aid generalization to novel concepts not seen during training. We also extend the task to include clarification questions.

Also related is work on interactive image retrieval such as allowing a user to mark relevant and irrelevant results (Nastar, Mitschke, and Meilhac 1998; Tieu and Viola 2004), which acts as a form of clarification. Recent works allow users to provide additional feedback using language to refine search results (Guo et al. 2018; Bhattacharya, Chowdhury, and Raykar 2019; Saha, Khapra, and Sankaranarayanan 2018). These directions are complementary to our work and could potentially be combined with it in the future.

3 Task Setup

We consider an interactive task of retrieving an image of a product based on a natural-language description. Given a set of candidate images and a description, the goal is to identify the image being referred to. Before trying to identify the target, the system can ask a combination of both clarification and active learning questions. The goal is to maximize the number of correct product identifications across interactions, while also keeping dialogs as short as possible.

Since we want to ensure that active learning questions are used to learn a generalizable classifier, we follow the setup of (Padmakumar, Stone, and Mooney 2018) and in each interaction we present the system with two sets of images:

- An active test set $\mathbf{I}^{Te}$ consisting of the candidate images to which the description could refer.
- An active training set $\mathbf{I}^{Tr}$ which is the set of images that can be queried for active learning.

It is also presented with a description of the target image. Before attempting to identify the target, the system can ask clarification or active learning questions. We assume the system has access to a set of attributes W that can be used in natural language descriptions of products. Given these attributes, the types of questions the system can ask are as follows (see Figure 1 for examples of each):

- Clarification query - A yes/no query about whether an attribute $w \in W$ is applicable to the target.
- Label query: A yes/no query about whether an attribute $w \in W$ is applicable to a specific image i in the active training set $\mathbf{I}^{Tr}$.
- Example query: Ask for a positive example in the active training set $\mathbf{I}^{Tr}$ for an attribute $w \in W$.

The dialog ends when the system makes a guess about the identity of the target, and is considered successful if it is correct. As in (Padmakumar, Stone, and Mooney 2018), we allow label and example queries that are either *on-topic* (queries about attributes in the current description) or *opportunistic* (queries that are not relevant to the current description but may be useful for future interactions), which have been shown to help interactive object retrieval (Thomason et al. 2017) (see Figure 1 for examples of each).

4 Methodology

4.1 Visual Attribute Classifier

We train a multilabel classifier for predicting visual attributes given an image. The network structure for the classifier is shown in Figure 2. We extract features $\phi(i)$ for the images using the penultimate layer of an Inception-V3 network (Szegedy et al. 2016) pretrained on ImageNet (Russakovsky et al. 2015). These are passed through two separate fully connected (FC) layers with ReLU activations, that are summed to produce the final representation $f(i)$ used for classification. This is converted into per-class probabilities $p(i)$ using a sigmoid layer with temperature correction (Guo et al. 2017). We obtain another set of per-class probabilities $p'(i)$ by passing the one of the intermediate representations $\psi'(i)$ through a sigmoid layer with temperature correction. Mathematically, given features $\phi(i)$ for image i, we have:

$$\begin{aligned} \psi(i) &= ReLU(w^T\phi(i) + b) \\ \psi'(i) &= ReLU(w'^T\phi(i) + b') \\ f(i) &= \psi(i) + \psi'(i) \end{aligned} \qquad \begin{aligned} p(i) &= \sigma(f(i) \odot \frac{1}{\tau}) \\ p'(i) &= \sigma(\psi'(i) \odot \frac{1}{\tau'}) \end{aligned}$$

where w, w', τ and τ' are learned vectors and b and b' are learned biases.

We train the network using a loss function that combines cross-entropy loss on $p(i)$ over all examples with the cross entropy loss over $p'(i)$ only for positive labels. That is,

$$\begin{aligned} L \quad = \quad & (1-\lambda)\sum_i y_i \log p(i) + (1-y_i)\log(1-p(i)) \\ & +\lambda \sum_i y_i \log p'(i) \end{aligned}$$

where y_i is the label vector for image i. This forces part of the network to focus on positive examples for each class (attribute). This is required because we use a heavily imbalanced dataset where most attributes have very few positive examples. We find this more effective than a standard weighted cross entropy loss, and the results in this paper use $\lambda = 0.9$. We also maintain a validation set of images labeled with attributes, that can be extended using active learning queries. Using this, we can estimate per-attribute precision, recall and F1. These metrics are used for tuning classifier hyperparameters and for dialog policy learning. More details about the classifier design are included in appendix **??**.

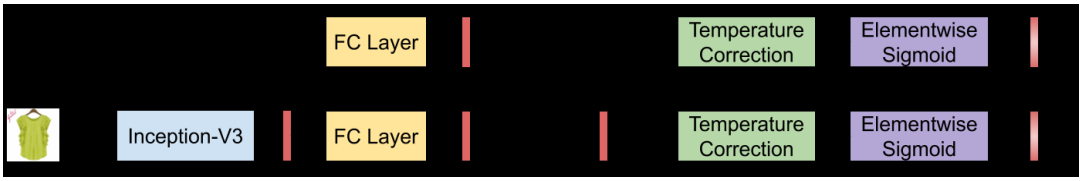


Figure 2: Visual Attribute Classifier

4.2 Grounding Model

We assume that a description is a conjunction of attributes, and use string-matching heuristics to determine the set of attributes referenced by the natural language description. Let the subset of attributes in the description d be $W_d \subseteq W$.

Suppose we additionally obtain from clarifications that attributes $W_p \subseteq W$ apply to the target image, and attributes $W_n \subseteq W$ do not apply, assuming independence of attributes, the probability that i is the target image, $b(i)$ is:

$$b(i) \propto \prod_{w \in W_d} p_w(i) \prod_{w \in W_p} p_w(i) \prod_{w \in W_n} (1 - p_w(i)) \quad (1)$$

At any stage, the best guess the system can make is the image with max belief, that is:

$$i_{guess} = \text{argmax}_{i \in \mathbf{I}^{Te}} b(i) \quad (2)$$

Also, we estimate the information gain J of a clarification $q \in W$ as follows. This is based on the formulation used in (Lee, Heo, and Zhang 2018) but we additionally make a Markov assumption (details in appendix **??**).

$$J(q) = \sum_{i \in \mathbf{I}^{Te}} \sum_{a \in \{0,1\}} b(i) P(a|q,i) \ln \left(\frac{P(a|q,i)}{P(a|q)} \right)$$

where $P(1|q,i) = p_q(i)$ and $P(0|q,i) = 1 - p_q(i)$ and $P(a|q) = \sum_i b(i) P(a|q,i)$.

4.3 MDP Formulation

We model each interaction as an episode in a Markov Decision Process (MDP) where the state consists of the images in the active training and test sets, the attributes mentioned in the target description, the current parameters of the classifier, and the set of queries asked and their responses. At each state, the agent has the following available actions:

- A special action for guessing – the image is chosen using Equation 2.
- One clarification query per attribute.
- A set of actions corresponding to possible active learning queries – one example query per attribute and one label query for each pair (w, i) for $w \in W$, $i \in \mathbf{I}^{Tr}$.

We do not allow actions to be repeated. We learn a hierarchical dialog policy composed of 3 parts – clarification and active learning policies to respectively choose the best clarification and active learning query in the current state, and a decision policy to choose between clarification, active learning, and guessing. An episode ends either when the guess action is chosen or a dialog length limit is reached, at which point the system is forced to make a guess. If the episode ends with a correct guess, the agent gets a large positive reward. Otherwise it gets a large negative reward at the end of the episode. Additionally, we use a small negative reward for each query to encourage shorter dialogs. In our experiments, we treat these rewards as tunable hyperparameters.

4.4 Policy Learning

We experimented with using both Q-learning and A3C (Mnih et al. 2016) for policy learning, both trained to maximize the discounted reward. Since the classifier has a large number of parameters, it is necessary to extract task-relevant features to represent state-action pairs. The features provided to each policy need to capture information from the current state that enable the system to identify useful clarifications and active learning queries, and trade off between these and guessing. The features used include:

4.5 Clarification policy features

- Metrics about the current beliefs $\{b(i) : i \in \mathbf{I}^{Te}\}$ and what they would be for each possible answer, if the question were asked:
  - Entropy: A higher entropy suggests that the agent is more uncertain. A decrease in entropy could indicate a good clarification.
  - Top two highest beliefs and their difference: A high value of the maximum belief, or a high difference between the top two beliefs could indicate that the agent is more confident about its guess. An increase in these could indicate a good clarification.
  - Difference between the maximum and average beliefs: A large difference suggests that the agent is more confident about its guess. An increase in these could indicate a good clarification.
- Information gain of the query as calculated in section 4.2.
- Current F1 of the attribute associated with the query: The system is likely to make better clarifications using attributes with high predictive accuracy.

4.6 Active learning policy features

- Current F1 of the attribute associated with the query, since the system is likely to benefit more from improving an attribute whose current predictive accuracy is not high.
- Fraction of previous dialogs in which the attribute has been used, since it is beneficial to focus on frequently used attributes that will likely benefit future dialogs.
- Fraction of previous dialogs using the attribute that have been successful, since this suggests that the attribute may be modelled well enough already.

- Whether the query is off-topic (i.e. opportunistic), since this would not benefit the current dialog.

Additionally, for label queries we use the following features:

- For query (w, i), $|p_w(i) - 0.5|$ as a measure of (un)certainty.
- Average cosine distance of the image to others in the dataset; this is motivated by density weighting to avoid selecting outliers.
- Fraction of k-nearest neighbors of the image that are unlabelled for this attribute, since a higher value suggests that the query could benefit multiple images.

4.7 Decision policy features

- Features of the current belief as in Sec. 4.5. These can help determine whether a guess is likely to be successful.
- Information gain of the best clarification action – to decide the utility of the clarification.
- Margin from the best active learning query if it is a label query – to decide the utility of the label query.
- F1 of attributes in clarification and active learning queries. High F1 is desirable for clarification and low F1 for active learning.
- Mean F1 of attributes in the description. A high value suggests that the belief is more reliable.
- Number of dialog turns completed.

4.8 Baseline static policy

As a baseline, we use an intuitive manually-designed static policy that is also hierarchical and was tailored to perform well in preliminary experiments. The static clarification policy chooses the attribute (among those with $F1 > 0$) with maximum information gain, with ties broken using F1. The static active learning policy has a fixed probability of choosing label queries and example queries. Uncertainty sampling is used to select the label query (w, i) with minimum $|p_w(i) - 0.5|$. An example query is chosen uniformly at random from the candidates. The decision policy initially chooses clarification if the information gain is above a minimum threshold, and the highest belief is below a confidence threshold. After a maximum number of clarifications, it chooses active learning until another threshold on the dialog length before guessing.

5 Experimental Setup

5.1 Dataset

To address a potential shopping application, we simulate dialogs using the iMaterialist Fashion Attribute data (Guo et al. 2019), consisting of images from the shopping site Wish[1] annotated for a set of 228 attributes. We scraped product descriptions for the images in the train and validation splits of the dataset for which attribute annotations are publicly available. After removing products whose images or descriptions were unavailable, we had 648,288 images with associated descriptions and attribute annotations.

We create a new data split following the protocol of (Padmakumar, Stone, and Mooney 2018) to ensure that the learned dialog policy generalizes to attributes not seen during policy training. We divided the data into 4 splits, *policy_pretrain*, *policy_train*, *policy_val* and *policy_test*, such that each contains images that have attributes for which positive examples are not present in earlier splits to increase the potential benefit of active learning. While we did not explicitly try to ensure that clarifications were beneficial, we validated that if we chose clarifications using an oracle that tries every clarification and selects the one that maximally increases the belief of the target image, it is possible to obtain a retrieval success rate of 80-85% without performing any active learning. Each of these is then split into subsets *classifier_training* and *classifier_test* by a uniform 60-40 split. More details are available in appendix **??**.

The *policy_pretrain* data is used to pretrain the multi-class attribute classifier. We use its *classifier_training* subset of images for training and its *classifier_test* subset to tune hyperparameters. The *policy_train* data is then used to learn the dialog policy. The *policy_val* data is used to tune hyperparameters as well as choose between RL algorithms. Finally, results are reported for the *policy_test* data.

We simulate dialogs as refinements of an initial retrieval based on the product description (details in appendix **??**). For the description of each image in the current *classifier_test* subset, we rank all other images in this subset according to a simplified version of the score in equation 1. From the images that get ranked within the top 1000 for their corresponding description, we sample target images for each interaction. The active test set for the interaction consists of the top 1000 images as ranked for that description. We randomly sample 1000 images from the appropriate *classifier_training* subset to form the active training set.

As in (Padmakumar, Stone, and Mooney 2018), we start the simulation of a dialog by providing the description of the target image to the agent. The annotated attributes are then used to automatically answer its queries and assess dialog success.

5.2 Experiment Phases

We run dialogs in batches of 100 and update the classifier and policies at the end of each batch. This is followed by repeating the retrieval step for all descriptions in the *classifier_test* subset before choosing target images for the next batch of dialogs. The experiment has the following phases:

- Classifier pretraining: We pretrain the classifier using annotated attribute labels for images in the *classifier_training* subset of the *policy_pretrain* set. This ensures that we have some reasonable clarifications at the start of dialog policy learning.
- Policy initialization: We initialize the dialog policy using experience collected using the baseline static policies (section 4.8) for the decision and active learning policies, and an oracle [2] to choose clarifications. This is done to

[1] https://www.wish.com/

[2] The oracle tries each candidate clarification and returns the one

Table 1: Results from the final batch of the test phase.

Decision Policy Type	Clarification Policy Type	Active Learning Policy Type	Fraction of Successful Dialogs	Average Dialog Length
Q-Learning	A3C	A3C	**0.33**	9.40
Q-Learning	A3C	Static	0.15	14.16
Q-Learning	Static	A3C	0.09	1.00
Static	A3C	A3C	0.27	20.00
Static	Static	Static	0.17	20.00

speed up policy learning. The dialogs for this phase are sampled from the set of *policy_train* images.

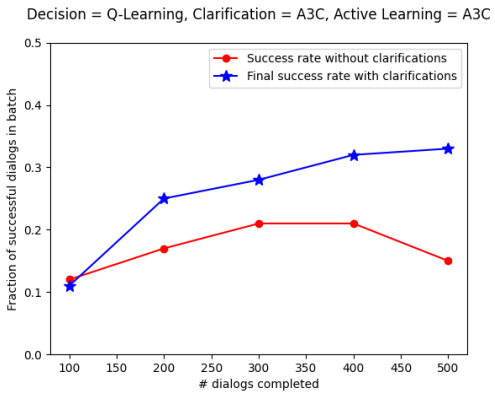
- Policy training: This phase consists of training the policy using on-policy experience, with dialogs again sampled from the set of *policy_train* images.
- Policy testing: We reset the classifier to the state at the end of pretraining. This is done to ensure that any performance improvement seen during testing are due to queries made during the testing phase. This is needed both for fair comparison with the baseline and to confirm that the system can generalize to novel attributes not seen during *any* stage of training. Dialogs are sampled for this from the *policy_val* set for hyperparameter tuning and from the *policy_test* set for reported results.

6 Results and Discussion

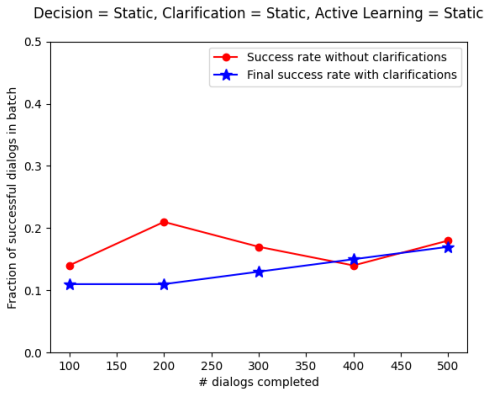
We initialize the policy with 4 batches of dialogs, followed by 4 batches of dialogs for the training phase, and 5 batches of dialogs in the testing phase. We compare the fully learned policy with hierarchical policies that consist of keeping one or more of the components static. We also compare the choice of Q-Learning or A3C (Mnih et al. 2016) as the policy learning algorithm for each learned policy. Table 1 shows the performance in the final test batch of the best fully learned policy, as well as a selected subset of the baselines (all conditions are included in appendix **??**). We evaluate policies on the fraction of successful episodes in the final test batch, and the average dialog length.

Ideally, we would like the system to have a high dialog success rate while having as low a dialog length as possible. We observe that using a learned policy for all three functions results in a significantly more successful dialog system (according to an unpaired Welch t-test with $p < 0.05$) than most conditions in which one or more of the policies are static. The exception is the case when the decision policy is static and the clarification and active learning policies are learned, in which case the difference is not statistically significant. The fully learned policy also uses significantly shorter dialogs than all conditions with a static decision policy. Some other conditions result in shorter dialogs, but these are unable to exploit the clarification and active learning actions enough to result in a success rate comparable to the fully learned policy.

that maximally increases the belief of the target image.



(a) Best Learned Policy



(b) Baseline Static Policy

Figure 3: Comparison of guess success rate with, and without clarifications across test batches.

Figure 3 plots the success rate across test batches, and the expected success rate if the system was forced to guess without clarification, for the fully learned, and fully static policies. These can be used to determine the individual effects of clarification and active learning. A significant increase in the success rate without clarification from the first to the final test batch suggests that active learning by itself is improving the retrieval ability of the system. We do not find this to be the case, either for the fully static or the fully learned policy. This shows that neither the static active learning policy nor the learned active learning policy are able to improve performance in the absence of clarification.

A significant increase from the success rate without clarification to the success rate with clarification would demonstrate that clarification is beneficial for any particular test batch. When performed on the first test batch, this demonstrates the effectiveness of clarifications alone, and when performed on the last test batch, this demonstrates the combined benefit of the clarification and active learning policies.

In the case of the fully static policy, we find that there is no statistically significant improvement, either in the expected initial success rate without clarifications, or in the final success rate, between the first and last test batch. This

suggests that neither the static active learning policy, nor its combination with the static clarification policy are capable of improving the system's performance.

However, in the case of the fully learned policy, we observe a statistically significant improvement in the final success rate, but not the initial success rate without clarifications. This suggests that while a learned active learning policy alone is not sufficient to improve the success rate, the *combination* of learned active learning and clarification policies *is* sufficient to improve the success rate. We also observe that while the difference between the initial and final success rate is initially not significant, it increases across batches, and becomes significant in the last two batches. This suggests that the clarification policy by itself is also insufficient for improvement, and the combination of the two is required to improve the system's success rate.

We believe that the reason for the relatively poor performance of the static clarification and active learning policies is that the classifier is not sufficiently accurate, and does not produce well calibrated probabilities, due to the heavy imbalance in the dataset. However, the learned policies are able to learn to properly adjust for this miscalibration.

7 Human Evaluation

We also compared our best learned policy and the baseline static policy using crowdworkers on Amazon Mechanical Turk. We had to make two important changes to our experiment setup for this evaluation. We had to remove images from the iMaterialist dataset that may require content warnings when shown to crowdworkers using a manually curated list of 24 attributes. We also found using pilot studies that workers on Amazon Mechanical Turk typically mentioned fewer of our annotated attributes than what is common in product descriptions, resulting in very low success rates from both policies. To account for this, we made the retrieval task easier by having the dialog agent select the target image from a set of 100 random images, and sampled a subset of the attributes from the product description during training. We then retrained the learned policy for the modified task setup with 10 batches of initialization, 10 batches of training and 5 batches of testing in simulation. We then used the final policy, and classifiers at the end of the test phase in interactions with users on Amazon Mechanical Turk to evaluate how well the learned system transfers. An image of the interface seen by the workers and some additional details are included in appendix **??**.

To minimize confusion, we presented each step of the dialog on a new page, and provided the users with visual examples of attributes used in the questions. We required workers to have completed at least 1000 HITs and have at least a 95% approval rate on their previous HITs, as well as complete a qualification task to demonstrate that they understood the types of questions used in our experiment. We had 50 workers interact with each system tested.

The results are shown in Table 2. We report the fraction of successful dialogs (in which the system guesses the correct target image), and average dialog length. Firstly, we observe that in this condition, the improvement in the learned policy in simulation is considerably higher than the static policy.

Table 2: Results of the static and new learned policies at the end of the test phase in simulation and in interactions on Amazon Mechanical Turk. Bold indicates a statistically significant improvement over the baseline ($p < 0.05$) and italic indicates trending significance ($p <= 0.1$) according to an unpaired Welch t-test.

Policy	Simulation – Fraction of Successful Dialogs	Simulation – Average Dialog Length	AMT – Fraction of Successful Dialogs	AMT – Average Dialog Length
Static	0.23	20.0	0.06	19.16
Learned	**0.65**	20.0	*0.16*	18.86

However, its average dialog length does not decrease. Qualitatively, we observe that in the test phase, the learned policy initially has a low rate of clarification, and high rate of active learning, which is reversed as the dialogs progress. We also observe that clarification is significantly more beneficial in this setting. We also observe a considerable drop in success rate for both the static and learned policy, in human interactions. However, the learned policy remains more successful than the static policy, trending towards significance ($p <= 0.1$) according to an unpaired Welch t-test.

We speculate that the drop in performance may be because of differences between the labels in the original dataset, and those provided by workers, making our classifiers less effective. We believe this is because some attributes are difficult for crowdworkers to label based on visual information alone, and we observed disagreement even on attributes we expected to be simpler, such as colors.

8 Conclusion

We demonstrate how a combination of RL learned policies for choosing attribute-based clarification and active learning queries can be used to improve an interactive system that needs to retrieve images based on a natural language description, while encountering novel attributes at test time not seen during training. Our experiments show that in challenging datasets where it is difficult to obtain an accurate attribute classifier, learned policies for choosing clarification and active learning queries outperform strong static baselines. We further show that in this challenging setup, a combination of learned clarification and active learning policies is necessary to obtain improvement over directly performing retrieval without interaction.

9 Future Work

In future, we would like to expand to free form natural language questions and active learning examples provided in other forms such as task demonstrations. Other directions we would like to explore are techniques to improve the sample efficiency of the active learning methods involved, as well as few-shot adaptation of better pretrained grounding models, to increase the gains from performing active learning.

Acknowledgements

We would like to thank Peter Stone, Joydeep Biswas and the UT Austin BWI group for helpful discussions. We would also like to thank the anonymous reviewers for their feedback. This work was supported by a Google Faculty Award received by Raymond J. Mooney and NSF NRI grants IIS-1925082 and IIS-1637736.

Broader Impact

Natural language interfaces such as language-based search, and intelligent personal assistants have the potential to make various forms of technology ranging from mobile phones and computers, as well as robots or other machines such as ATMs or self-checkout counters more accessible and less intimidating to users who are unfamiliar or uncomfortable with other interfaces on such devices such as command shells, button based interfaces or changing visual user interfaces. Spoken language interfaces can also be used to make such devices more accessible for the visually impaired or users who have difficulty with fine motor control.

However, the use of these interfaces do involve concerns over privacy and data security. This is especially the case with devices based on spoken language interfaces as they need to analyze every conversation for potential codewords (Lackes, Siepermann, and Vetter 2019). Thus, users need to trust that these extraneous conversations will not be stored, or analyzed for other information. This is particularly problematic in environments such as hospitals or lawyer's officers where confidentiality is expected.

Another concern is that transactions on these devices may be triggered by casual conversation or voices on television (Liptak 2017), that were not intended to activate the dialog system. A related concern is that the ambiguity of language or mistakes made by the system may trigger unintended actions. In most applications, these can be handled by setting up appropriate confirmation or cancellation procedures for sensitive actions. Increased use of clarification steps before execution of an action may provide an additional opportunity for users to cancel such actions before they take place.

Using active learning, or any form of continuous learning with user data can make machine learning systems more useful due to increased exposure to the data distribution with which such systems need to operate in practice. However, most machine learning algorithms assume that the input data is complete and correct, both of which may be violated by systems that train on user-generated data. It is also possible for such data to be biased in a variety of ways – ranging from potential absence of representation or misrepresentation of some groups of people who do not use the system as frequently, to filter-bubble like effects when many users provide a few frequent examples as training data to the system (Baeza-Yates 2016). Explicit active learning questions also allow users to deliberately provide misinformation to machine learning systems. Practical systems using active learning need to incorporate methods for handling noisy data, and need to have tests in place for undesirable learned biases.

References

Bachman, P.; Sordoni, A.; and Trischler, A. 2017. Learning Algorithms for Active Learning. In *ICML*, 301–310.

Baeza-Yates, R. 2016. Data and Algorithmic Bias in the Web. In *ACM Conference on Web Science*, 1–1.

Bhattacharya, I.; Chowdhury, A.; and Raykar, V. C. 2019. Multimodal Dialog for Browsing Large Visual Catalogs Using Exploration-Exploitation Paradigm in a Joint Embedding Space. In *MMR*, 187–191.

Bordes, A.; Boureau, Y.-L.; and Weston, J. 2017. Learning End-to-End Goal-Oriented Dialog. In *ICLR*.

Budzianowski, P.; Ultes, S.; Su, P.-H.; Mrkšić, N.; Wen, T.-H.; Casanueva, I.; Barahona, L. M. R.; and Gasic, M. 2017. Sub-domain Modelling for Dialogue Management with Hierarchical Reinforcement Learning. In *SIGDIAL*, 86–92.

Casanueva, I.; Budzianowski, P.; Su, P.-H.; Ultes, S.; Rojas-Barahona, L.; Tseng, B.-H.; and Gašic, M. 2018. Feudal Reinforcement Learning for Dialogue Management in Large Domains. In *Proceedings of NAACL-HLT*, 714–719.

Chaurasia, S.; and Mooney, R. J. 2017. Dialog for Language to Code. In *IJCNLP*, 175–180.

De Vries, H.; Strub, F.; Chandar, S.; Pietquin, O.; Larochelle, H.; and Courville, A. 2017. GuessWhat?! Visual Object Discovery Through Multi-modal Dialogue. In *CVPR*.

Deits, R.; Tellex, S.; Thaker, P.; Simeonov, D.; Kollar, T.; and Roy, N. 2013. Clarifying Commands with Information-theoretic Human-robot Dialog. *Journal of Human-Robot Interaction* 2(2): 58–79.

Dindo, H.; and Zambuto, D. 2010. A Probabilistic Approach to Learning a Visually Grounded Language Model Through Human-Robot Interaction. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 790–796. IEEE.

Dodge, J.; Gane, A.; Zhang, X.; Bordes, A.; Chopra, S.; Miller, A.; Szlam, A.; and Weston, J. 2016. Evaluating Prerequisite Qualities for Learning End-to-end Dialog Systems. In *ICLR*.

Fang, M.; Li, Y.; and Cohn, T. 2017. Learning how to Active Learn: A Deep Reinforcement Learning Approach. In *EMNLP*.

Farhadi, A.; Endres, I.; Hoiem, D.; and Forsyth, D. 2009. Describing Objects by their Attributes. In *CVPR*, 1778–1785.

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On Calibration of Modern Neural Networks. In *ICML*, 1321–1330.

Guo, S.; Huang, W.; Zhang, X.; Srikhanta, P.; Cui, Y.; Li, Y.; Adam, H.; Scott, M. R.; and Belongie, S. 2019. The iMaterialist Fashion Attribute Dataset. In *ICCV Workshops*, 0–0.

Guo, X.; Wu, H.; Cheng, Y.; Rennie, S.; Tesauro, G.; and Feris, R. 2018. Dialog-based Interactive Image Retrieval. In *NeurIPS*, 678–688.

Hu, H.; Wu, X.; Luo, B.; Tao, C.; Xu, C.; Wu, W.; and Chen, Z. 2018. Playing 20 Question Game with Policy-Based Reinforcement Learning. In *EMNLP*.

Lackes, R.; Siepermann, M.; and Vetter, G. 2019. Can I Help You?–The Acceptance of Intelligent Personal Assistants. In *International Conference on Business Informatics Research*, 204–218.

Lee, S.-W.; Heo, Y.-J.; and Zhang, B.-T. 2018. Answerer in Questioner's Mind for Goal-oriented Visual Dialogue. In *Visually-Grounded Interaction and Language Workshop (NeurIPS)*.

Li, Y.; Huang, C.; Tang, X.; and Change Loy, C. 2017. Learning to Disambiguate by Asking Discriminative Questions. In *ICCV*, 3419–3428.

Liptak, A. 2017. Amazon's Alexa started ordering people dollhouses after hearing its name on TV. URL https://www.theverge.com/2017/1/7/14200210/amazon-alexa-tech-news-anchor-order-dollhouse.

Manuvinakurike, R.; DeVault, D.; and Georgila, K. 2017. Using reinforcement learning to model incrementality in a fast-paced dialogue game. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 331–341.

Mnih, V.; Badia, A. P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; and Kavukcuoglu, K. 2016. Asynchronous Methods for Deep Reinforcement Learning. In *ICML*, 1928–1937.

Nastar, C.; Mitschke, M.; and Meilhac, C. 1998. Efficient Query Refinement for Image Retrieval. In *CVPR*, 547–552.

Padmakumar, A.; Stone, P.; and Mooney, R. J. 2018. Learning a Policy for Opportunistic Active Learning. In *EMNLP*.

Padmakumar, A.; Thomason, J.; and Mooney, R. J. 2017. Integrated Learning of Dialog Strategies and Semantic Parsing. In *EACL*, 547–557.

Parde, N.; Hair, A.; Papakostas, M.; Tsiakas, K.; Dagioglou, M.; Karkaletsis, V.; and Nielsen, R. D. 2015. Grounding the Meaning of Words Through Vision and Interactive Gameplay. In *IJCAI*.

Peng, B.; Li, X.; Li, L.; Gao, J.; Celikyilmaz, A.; Lee, S.; and Wong, K.-F. 2017. Composite Task-Completion Dialogue Policy Learning via Hierarchical Deep Reinforcement Learning. In *EMNLP*, 2231–2240.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *IJCV* 115(3): 211–252. doi: 10.1007/s11263-015-0816-y. URL https://doi.org/10.1007\%2Fs11263-015-0816-y.

Saha, A.; Khapra, M. M.; and Sankaranarayanan, K. 2018. Towards Building Large scale Multimodal Domain-Aware Conversation Systems. In *AAAI*.

Settles, B. 2010. Active Learning Literature Survey. *University of Wisconsin, Madison* 52(55-66): 11.

Strub, F.; De Vries, H.; Mary, J.; Piot, B.; Courvile, A.; and Pietquin, O. 2017. End-to-end Optimization of Goal-driven and Visually Grounded Dialogue Systems. In *IJCAI*, 2765–2771.

Su, P.-H.; Gasic, M.; Mrksic, N.; Rojas-Barahona, L. M.; Ultes, S.; Vandyke, D.; Wen, T.-H.; and Young, S. J. 2016. On-line Active Reward Learning for Policy Optimisation in Spoken Dialogue Systems. In *ACL (1)*.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the Inception Architecture for Computer Vision. In *CVPR*, 2818–2826.

Thomason, J.; Padmakumar, A.; Sinapov, J.; Hart, J.; Stone, P.; and Mooney, R. J. 2017. Opportunistic Active Learning for Grounding Natural Language Descriptions. In *CoRL*, 67–76.

Thomason, J.; Zhang, S.; Mooney, R.; and Stone, P. 2015. Learning to Interpret Natural Language Commands through Human-Robot Dialog. In *IJCAI*, 1923–1929.

Tieu, K.; and Viola, P. 2004. Boosting Image Retrieval. *IJCV* 56(1-2): 17–36.

Wen, T.-H.; Gašić, M.; Mrkšić, N.; Rojas-Barahona, L. M.; Su, P.-H.; Ultes, S.; Vandyke, D.; and Young, S. 2016. A Network-based End-to-End Trainable Task-oriented Dialogue System . In *NAACL*.

Williams, J.; Raux, A.; and Henderson, M. 2016. The Dialog State Tracking Challenge Series: A Review. *Dialogue & Discourse* 7(3): 4–33.

Woodward, M.; and Finn, C. 2017. Active one-shot learning. *Computing Research Repository* arXiv:1702.06559.

Young, S.; Gašić, M.; Thomson, B.; and Williams, J. D. 2013. POMDP-based Statistical Spoken Dialog Systems: A Review. *Proceedings of the IEEE* 101(5): 1160–1179.

Yu, Y.; Eshghi, A.; and Lemon, O. 2017. Learning how to Learn: An Adaptive Dialogue Agent for Incrementally Learning Visually Grounded Word Meanings . In *Workshop on Language Grounding for Robotics*.

Zhang, J.; Wu, Q.; Shen, C.; Zhang, J.; Lu, J.; and Van Den Hengel, A. 2018. Goal-oriented Visual Question Generation via Intermediate Rewards. In *ECCV*, 186–201.

Zhang, J.; Zhao, T.; and Yu, Z. 2018. Multimodal Hierarchical Reinforcement Learning Policy for Task-Oriented Visual Dialog. In *SIGDIAL*, 140–150.

Zhu, Y.; Zhang, S.; and Metaxas, D. 2017. Interactive Reinforcement Learning for Object Grounding via Self-talking. *Computing Research Repository* arXiv:1712.00576.