

Training an adaptive dialogue policy for interactive learning of visually grounded word meanings

Yanchao Yu
Interaction Lab
Heriot-Watt University
y.yu@hw.ac.uk

Arash Eshghi
Interaction Lab
Heriot-Watt University
a.eshghi@hw.ac.uk

Oliver Lemon
Interaction Lab
Heriot-Watt University
o.lemon@hw.ac.uk

Abstract

We present a multi-modal dialogue system for interactive learning of perceptually grounded word meanings from a human tutor. The system integrates an incremental, semantic parsing/generation framework - Dynamic Syntax and Type Theory with Records (DS-TTR) - with a set of visual classifiers that are learned throughout the interaction and which ground the meaning representations that it produces. We use this system in interaction with a simulated human tutor to study the effects of different dialogue policies and capabilities on accuracy of learned meanings, learning rates, and efforts/costs to the tutor. We show that the overall performance of the learning agent is affected by (1) who takes initiative in the dialogues; (2) the ability to express/use their confidence level about visual attributes; and (3) the ability to process elliptical and incrementally constructed dialogue turns. Ultimately, we train an adaptive dialogue policy which optimises the trade-off between classifier accuracy and tutoring costs.

1 Introduction

Identifying, classifying, and talking about objects or events in the surrounding environment are key capabilities for intelligent, goal-driven systems that interact with other agents and the external world (e.g. robots, smart spaces, and other automated systems). To this end, there has recently been a surge of interest and significant progress made on a variety of related tasks, including generation of Natural Language (NL) descriptions of images, or identifying images based on NL descriptions (Karpathy and Fei-Fei, 2014; Bruni et

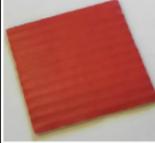
Dialogue Example		Final Semantics in TTR
T: what is this? S: a red circle? T: no, a red square. S: oh, okay.		$\left(\begin{array}{l} X_{=o1} : e \\ p2 : \text{red}(X) \\ p3 : \text{square}(X) \end{array} \right)$
T: what can you see? S: something orange. T: what shape is it S: a square. T: no, it's a circle. S: uhu		$\left(\begin{array}{l} X_{1=o2} : e \\ S=s : \text{per} \\ p : \text{circle}(X1) \\ p1 : \text{orange}(X1) \\ p2 : \text{see}(S,X1) \end{array} \right)$

Figure 1: Example dialogues & interactively agreed semantic contents.

al., 2014; Socher et al., 2014; Farhadi et al., 2009; Silberer and Lapata, 2014; Sun et al., 2013).

Our goal is to build *interactive* systems that can learn grounded word meanings relating to their perceptions of real-world objects – this is different from previous work such as e.g. (Roy, 2002), that learn groundings from descriptions without any interaction, and more recent work using Deep Learning methods (e.g. (Socher et al., 2014)).

Most of these systems rely on training data of high quantity with no possibility of online error correction. Furthermore, they are unsuitable for robots and multimodal systems that need to continuously, and incrementally learn from the environment, and may encounter objects they haven't seen in training data. These limitations are likely to be alleviated if systems can learn concepts, as and when needed, from situated dialogue with humans. Interaction with a human tutor also enables systems to take initiative and seek the particular information they need or lack by e.g. asking questions with the highest information gain (see e.g. (Skocaj et al., 2011), and Fig. 1). For example, a robot could ask questions to learn the colour of a “square” or to request to be presented with more “red” things to improve its performance on the concept (see e.g. Fig. 1). Furthermore, such systems could allow for meaning negotiation in the

form of clarification interactions with the tutor.

This setting means that the system must be *trainable from little data, compositional, adaptive, and able to handle natural human dialogue with all its glorious context-sensitivity and messiness* – for instance so that it can learn visual concepts suitable for specific tasks/domains, or even those specific to a particular user. Interactive systems that learn continuously, and over the long run from humans need to do so *incrementally, quickly, and with minimal effort/cost to human tutors*.

In this paper, we first outline an implemented dialogue system that integrates an incremental, semantic grammar framework, especially suited to dialogue processing – Dynamic Syntax and Type Theory with Records (DS-TTR¹ (Kempson et al., 2001; Eshghi et al., 2012)) with visual classifiers which are learned during the interaction, and which provide perceptual grounding for the basic semantic atoms in the semantic representations (Record Types in TTR) produced by the parser (see Fig. 1, Fig. 2 and section 3).

We then use this system in interaction with a simulated human tutor, to test hypotheses about how the accuracy of learned meanings, learning rates, and the overall cost/effort for the human tutor are affected by different dialogue policies and capabilities: (1) who takes **initiative** in the dialogues; (2) the agent’s ability to utilise their level of **uncertainty** about an object’s attributes; and (3) their ability to process **elliptical as well as incrementally constructed dialogue turns**. The results show that differences along these dimensions have significant impact both on the accuracy of the grounded word meanings that are learned, and the processing effort required by the tutors.

In section 4.3 we train an adaptive dialogue strategy that finds a better trade-off between classifier accuracy and tutor cost.

2 Related work

In this section, we will present an overview of vision and language processing systems, as well as multi-modal systems that learn to associate them. We compare them along two main dimensions: *Visual Classification methods: offline vs. online* and *the kinds of representation learned/used*.

Online vs. Offline Learning. A number of implemented systems have shown good performance on classification as well as NL-description of

novel physical objects and their attributes, either using offline methods as in (Farhadi et al., 2009; Lampert et al., 2014; Socher et al., 2013; Kong et al., 2013), or through an incremental learning process, where the system’s parameters are updated after each training example is presented to the system (Furao and Hasegawa, 2006; Zheng et al., 2013; Kristan and Leonardis, 2014). For the interactive learning task presented here, only the latter is appropriate, as the system is expected to learn from its interactions with a human tutor over a period of time. Shen & Hasegawa (2006) propose the SOINN-SVM model that re-trains linear SVM classifiers with data points that are clustered together with all the examples seen so far. The clustering is done incrementally, but the system needs to keep all the examples so far in memory. Kristian & Leonardis (2014), on the other hand, propose the oKDE model that continuously learns categorical knowledge about visual attributes as probability distributions over the categories (e.g. colours). However, when learning from scratch, it is unrealistic to predefine these concept groups (e.g. that red, blue, and green are colours). Systems need to learn for themselves that, e.g. colour is grounded in a specific sub-space of an object’s features. For the visual classifiers, we therefore assume no such category groupings here, and instead learn individual binary classifiers for each visual attribute (see section 3.1 for details).

Distributional vs. Logical Representations.

Learning to ground natural language in perception is one of the fundamental problems in Artificial Intelligence. There are two main strands of work that address this problem: (1) those that learn distributional representations using Deep Learning methods: this often works by projecting vector representations from different modalities (e.g. vision and language) into the same space in order to be able to retrieve one from the other (Socher et al., 2014; Karpathy and Li, 2015; Silberer and Lapata, 2014); (2) those that attempt to ground symbolic logical forms, obtained through semantic parsing (Tellex et al., 2014; Kollar et al., 2013; Matuszek et al., 2014) in classifiers of various entities types/events/relations in a segment of an image or a video. Perhaps one advantage of the latter over the former method, is that it is strictly compositional, i.e. the contribution of the meaning of an individual word, or semantic atom, to the whole representation is clear, whereas this is hard to say

¹Download from <http://dylan.sourceforge.net>

about the distributional models. As noted, our work also uses the latter methodology, though it is dialogue, rather than sentence semantics that we care about. Most similar to our work is probably that of Kennington & Schlangen (2015) who learn a mapping between individual words - rather than logical atoms - and low-level visual features (e.g. colour-values) directly. The system is compositional, yet does not use a grammar (the compositions are defined by hand). Further, the groundings are learned from pairings of object references in NL and images rather than from dialogue.

What sets our approach apart from others is: a) that we use a domain-general, incremental semantic grammar with principled mechanisms for parsing and generation; b) Given the DS model of dialogue (Eshghi et al., 2015), representations are constructed jointly and interactively by the tutor and system over the course of several turns (see Fig. 1); c) perception and NL-semantics are modelled in a single logical formalism (TTR); d) we effectively induce an ontology of atomic types in TTR, which can be combined in arbitrarily complex ways for generation of complex descriptions of arbitrarily complex visual scenes (see e.g. (Dobnik et al., 2012) and compare this with (Kennington and Schlangen, 2015), who do not use a grammar and therefore do not have logical structure over grounded meanings).

3 System Architecture

We have developed a system to support an attribute-based object learning process through natural, incremental spoken dialogue interaction. The architecture of the system is shown in Fig. 2. The system has two main modules: a vision module for visual feature extraction, classification, and learning; and a dialogue system module using DS-TTR. Below we describe these components individually and then explain how they interact.

3.1 Attribute-based Classifiers used

Yu et. al (2015a; 2015b) point out that neither multi-label classification models nor ‘zero-shot’ learning models show acceptable performance on attribute-based learning tasks. Here, we instead use Logistic Regression SVM classifiers with Stochastic Gradient Descent (SGD) (Zhang, 2004) to incrementally learn attribute predictions.

All classifiers will output attribute-based label sets and corresponding probabilities for novel un-

seen images by predicting binary label vectors. We build visual feature representations to learn classifiers for particular attributes, as explained in the following subsections.

3.1.1 Visual Feature Representation

In contrast to previous work (Yu et al., 2015a; Yu et al., 2015b), to reduce feature noise through the learning process, we simplify the method of feature extraction consisting of two base feature categories, i.e. the colour space for colour attributes, and a ‘bag of visual words’ for the object shapes/class.

Colour descriptors, consisting of HSV colour space values, are extracted for each pixel and then are quantized to a $16 \times 4 \times 4$ HSV matrix. These descriptors inside the bounding box are binned into individual histograms. Meanwhile, a bag of visual words is built in PHOW descriptors using a visual dictionary (that is pre-defined with a hand-made image set). These visual words will be calculated using 2×2 blocks, a 4-pixel step size, and quantized into 1024 k-means centres. The feature extractor in the vision module presents a 1280-dimensional feature vector for a single training/test instance by stacking all quantized features, as shown in Figure 2.

3.2 Dynamic Syntax and Type Theory with Records

Dynamic Syntax (DS) is a word-by-word incremental semantic parser/generator, based around the Dynamic Syntax (DS) grammar framework (Cann et al., 2005) especially suited to the fragmentary and highly contextual nature of dialogue. In DS, dialogue is modelled as the interactive and incremental construction of contextual and semantic representations (Eshghi et al., 2015). The contextual representations afforded by DS are of the fine-grained semantic content that is jointly negotiated/agreed upon by the interlocutors, as a result of processing questions and answers, clarification requests, corrections, acceptances, etc. We cannot go into any further detail due to lack of space, but proceed to introduce Type Theory with Records, the formalism in which the DS contextual/semantic representations are couched, but also that within which perception is modelled here.

Type Theory with Records (TTR) is an extension of standard type theory shown to be use-

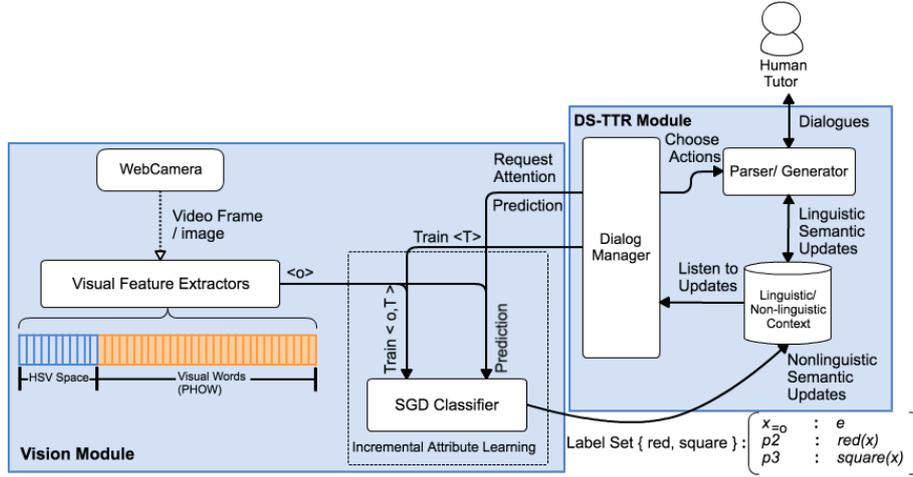


Figure 2: Architecture of the teachable system

ful in semantics and dialogue modelling (Cooper, 2005; Ginzburg, 2012). TTR is particularly well-suited to our problem here as it allows information from various modalities, including vision and language, to be represented within a single semantic framework (see e.g. Larsson (2013); Dobnik et al. (2012) who use it to model the semantics of spatial language and perceptual classification).

In TTR, logical forms are specified as *record types* (RTs), which are sequences of *fields* of the form $[l : T]$ containing a label l and a type T . RTs can be witnessed (i.e. judged true) by *records* of that type, where a record is a sequence of label-value pairs $[l = v]$. We say that $[l = v]$ is of type $[l : T]$ just in case v is of type T .

$$R_1 : \left[\begin{array}{l} l_1 : T_1 \\ l_2=a : T_2 \\ l_3=p(l_2) : T_3 \end{array} \right] \quad R_2 : \left[\begin{array}{l} l_1 : T_1 \\ l_2 : T_2' \end{array} \right] \quad R_3 : []$$

Figure 3: Example TTR record types

Fields can be *manifest*, i.e. given a singleton type e.g. $[l : T_a]$ where T_a is the type of which only a is a member; here, we write this using the syntactic sugar $[l=a : T]$. Fields can also be *dependent* on fields preceding them (i.e. higher) in the record type (see Fig. 3).

The standard subtype relation \sqsubseteq can be defined for record types: $R_1 \sqsubseteq R_2$ if for all fields $[l : T_2]$ in R_2 , R_1 contains $[l : T_1]$ where $T_1 \sqsubseteq T_2$. In Figure 3, $R_1 \sqsubseteq R_2$ if $T_2 \sqsubseteq T_2'$, and both R_1 and R_2 are subtypes of R_3 . This subtyping relation allows semantic information to be incrementally specified, i.e. record types can be indefinitely extended with more information/constraints. For us here, this is a key feature since it allows the system to en-

code *partial* knowledge about objects, and for this knowledge (e.g. object attributes) to be extended in a principled way, as and when this information becomes available.

3.3 Integration

Fig. 2 shows how the various parts of the system interact. At any point in time, the system has access to an ontology of (object) types and attributes encoded as a set of TTR Record Types, whose individual atomic symbols, such as ‘red’ or ‘square’ are grounded in the set of classifiers trained so far.

Given a set of individuated objects in a scene, encoded as a TTR Record, the system can utilise its existing ontology to output a Record Type which maximally characterises the scene (see e.g. Fig. 1). Dynamic Syntax operates over the same representations, they provide a direct interface between perceptual classification and semantic processing in dialogue: this representation acts not only as (1) the non-linguistic (here, visual) context of the dialogue for the resolution of e.g. definite reference and indexicals (see (Hough and Purver, 2014)); but also as (2) the logical database from which the system can generate utterances (descriptions), ask, or answer questions about the objects - Fig. 4 illustrates how the semantics of the answer to a question is retrieved from the visual context through *unification* (this uses the standard subtype checking operation within TTR).

Conversely, for concept learning, the DS-TTR parser incrementally produces Record Types (RT), representing the meaning jointly established by the tutor and the system so far. In this domain, this is ultimately one or more type judgements, i.e. that some scene/image/object is judged to be of a

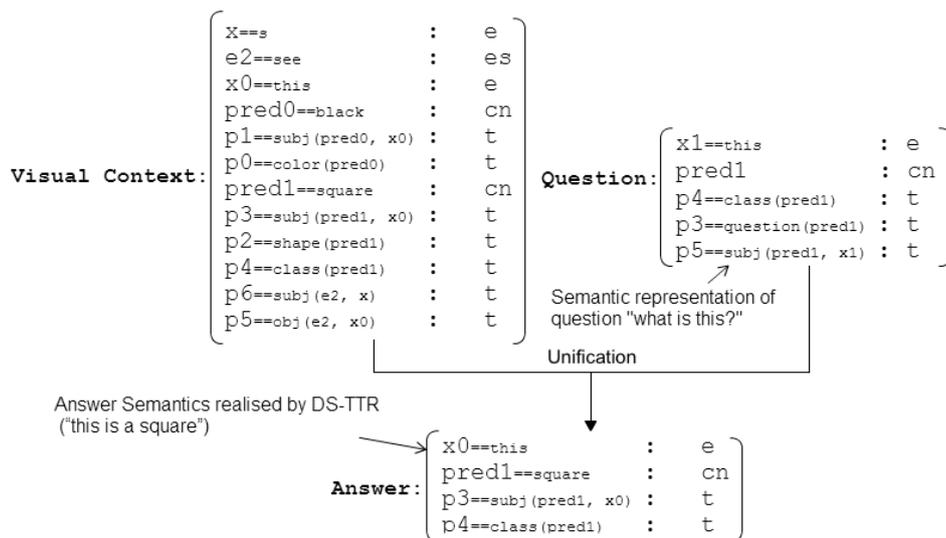


Figure 4: Question Answering by the system

particular type, e.g. in Fig. 1 that the individuated object, $o1$ is a red square. These jointly negotiated type judgements then go on to provide training instances for the classifiers. In general, the training instances are of the form, $\langle O, T \rangle$, where O is an image/scene segment (an object or TTR Record), and T , a record type. T is then decomposed into its constituent atomic types $T_1 \dots T_n$, s.t. $\bigwedge T_i = T$ - where \bigwedge is the so called *meet* operation corresponding to type conjunction. The judgements $O : T_i$ are then used directly to train the classifiers that ground the T_i .

4 Experimental Setup

As noted in the introduction, interactive systems that learn continuously, and over the long run from humans need to do so *incrementally*; *as quickly as possible*; and *with as little effort/cost to the human tutor as possible*. In addition, when learning takes place through dialogue, the dialogue needs to be as human-like/natural as possible.

In general, there are several different dialogue capabilities and policies that a concept-learning agent might adopt, and these will lead to different outcomes for the accuracy of the learned concepts/meanings, learning rates, and cost to the tutor – with trade-offs between these. Our goal in this paper is therefore an experimental study of the effect of different dialogue policies and capabilities on the overall performance of the learning agent, which, as we describe below is a measure capturing the trade-off between accuracy of learned meanings and the cost of tutoring.

Design. We use the dialogue system outlined above to carry out our main experiment with a $2 \times 2 \times 2$ factorial design, i.e. with three factors each with two levels. Together, these factors determine the learner’s dialogue behaviour: (1) **Initiative (Learner/Tutor)**: determines who takes initiative in the dialogues. When the tutor takes initiative, s/he is the one that drives the conversation forward, by asking questions to the learner (e.g. “What colour is this?” or “So this is a”) or making a statement about the attributes of the object. On the other hand, when the learner has initiative, it makes statements, asks questions, initiates topics etc. (2) **Uncertainty (+UC/-UC)**: determines whether the learner takes into account, in its dialogue behaviour, its own subjective confidence about the attributes of the presented object. The confidence is the probability assigned by any of its attribute classifiers of the object being a positive instance of an attribute (e.g. ‘red’) - see below for how a confidence threshold is used here. In +UC, the agent will not ask a question if it is confident about the answer, and it will hedge the answer to a tutor question if it is not confident, e.g. “T: What is this? L: errm, maybe a square?”. In -UC, the agent always takes itself to know the attributes of the given object (as given by its currently trained classifiers), and behaves according to that assumption. (3) **Context-Dependency (+CD/-CD)**: determines whether the learner can process (produce/parse) context-dependent expressions such as short answers and incrementally constructed turns, e.g. “T: What is this? L: a square”, or “T:

L-UC-CD L: This is red. T: No, it is blue. L: Okay. This is a square. T: Yes.	L+UC+CD L: What colour is this? T: Red. L: Okay. Is this a square? T: No, a circle. L: Okay.	L+UC-CD L: Is this a circle? T: No, it's a triangle. L: Okay. Is it green? T: Yes.	L-UC+CD L: This is a square T: No, a triangle. L: Okay. This is red. T: Yes.
T+UC+CD T: This is a ... L: Errm, a square? T: Yes. What colour is it? L: Red. T: No, it's green. L: Okay.	T-UC-CD T: What (shape) is this? L: This is a circle. T: Yes. What colour is it? L: it is red. T: No, it's purple. L: Okay.	T+UC-CD T: What is this? L: (long pause) T: It is a square. L: Okay. T: What colour is it? L: Is it blue? T: Yes.	T-UC+CD T: What is this? L: A square. T: Yes. What colour is it? L: Blue. T: No, it is green. L: Uhu.

Figure 5: Example dialogues in different conditions

So this one is ...? L: red/a circle”. This setting can be turned off/on in the DS-TTR dialogue model.

Tutor Simulation and Policy To run our experiment on a large-scale, we have hand-crafted an *Interactive Tutoring Simulator*, which simulates the behaviour of a human tutor². The tutor policy is kept constant across all conditions. Its policy is that of an always *truthful*, *helpful* and *omniscient* one: it (1) has complete access to the labels of each object; and (2) always acts as the context of the dialogue dictates: answers any question asked, confirms or rejects when the learner describes an object; and (3) always corrects the learner when it describes an object erroneously.

Dependent Measures We now go on to describe the dependent measures in our experiment, i.e. that of classifier accuracy/score, tutoring cost, and the overall performance measure which combines the former two measures.

Confidence Threshold To determine when the agent takes themselves to be confident in an attribute prediction, we use confidence-score thresholds. It consists of two values, a base threshold (e.g. 0.5) and a positive threshold (e.g. 0.9).

If the confidences of all classifiers are under the base threshold (i.e. the learner has no attribute label that it is confident about), the agent will ask for information directly from the tutor via questions (e.g. “L: what is this?”).

On the other hand, if one or more classifiers score above the base threshold, then the positive threshold is used to judge to what extent the agent

²The experiment involves hundreds of dialogues, so running this experiment with real human tutors has proven too costly at this juncture, though we plan to do this for a full evaluation of our system in the future.

trusts its prediction or not. If the confidence score of a classifier is between the positive and base thresholds, the learner is not very confident about its knowledge, and will check with the tutor, e.g. “L: is this red?”. However, if the confidence score of a classifier is above the positive threshold, the learner is confident enough in its knowledge not to bother verifying it with the tutor. This will lead to less effort needed from the tutor as the learner becomes more confident about its knowledge. However, since a learning agent that has high confidence about a prediction will not ask for assistance from the tutor, a low positive threshold would reduce the chances that allow the tutor to correct the learner’s mistakes. We therefore tested different fixed values for the confidence threshold and this determined a fixed 0.5 base threshold and a 0.9 positive threshold were deemed to be the most appropriate values for an interactive learning process - i.e. these values preserved good classifier accuracy while not requiring much effort from the tutor - see below Section 4.3 for how an adaptive policy was learned that adjusts the agent’s confidence threshold dynamically over time.

4.1 Evaluation Metrics

To test how the different dialogue capabilities and strategies affect the learning process, we consider both the cost to the tutor and the accuracy of the learned meanings, i.e. the classifiers that ground our colour and shape concepts.

Cost The cost measure reflects the effort needed by a human tutor in interacting with the system. Skocaj et. al. (2009) point out that a comprehensive teachable system should learn as autonomously as possible, rather than involving the human tutor too frequently. There are several pos-

Table 1: Tutoring Cost Table

C_{inf}	C_{ack}	C_{crt}	$C_{parsing}$	$C_{production}$
1	0.25	1	0.5	1

sible costs that the tutor might incur, see Table 1: C_{inf} refers to the cost of the tutor providing information on a single attribute concept (e.g. “this is red” or “this is a square”); C_{ack} is the cost for a simple confirmation (like “yes”, “right”) or rejection (such as “no”); C_{crt} is the cost of correction for a single concept (e.g. “no, it is blue” or “no, it is a circle”). We associate a higher cost with correction of statements than that of polar questions. This is to penalise the learning agent when it confidently makes a false statement – thereby incorporating an aspect of trust in the metric (humans will not trust systems which confidently make false statements). And finally, parsing (C_{parse}) as well as production ($C_{production}$) costs for tutor are taken into account: each single word costs 0.5 when parsed by the tutor, and 1 if generated (production costs twice as much as parsing). These exact values are based on intuition but are kept constant across the experimental conditions and therefore do not confound the results reported below.

Learning Performance As mentioned above, an efficient learner dialogue policy should consider both classification accuracy and tutor effort (Cost). We thus define an integrated measure – the *Overall Performance Ratio* (R_{perf}) – that we use to compare the learner’s overall performance across the different conditions:

$$R_{perf} = \frac{\Delta Acc}{C_{tutor}}$$

i.e. the increase in accuracy per unit of the cost, or equivalently the gradient of the curve in Fig. 4c. We seek dialogue strategies that maximise this.

Dataset The dataset used here is comprised of 600 images of single, simple handmade objects with a white background (see Fig.1)³. There are nine attributes considered in this dataset: 6 colours (black, blue, green, orange, purple and red) and 3 shapes (circle, square and triangle), with a relative balance on the number of instances per attribute.

4.2 Evaluation and Cross-validation

In each round, the system is trained using 500 training instances, with the rest set aside for test-

³All data from this paper will be made freely available.

ing. For each training instance, the system interacts (only through dialogue) with the simulated tutor. Each dialogue about an object ends either when both the shape and the colour of the object are discussed and agreed upon, or when the learner requests to be presented with the next image (this happens only in the Learner initiative conditions). We define a **Learning Step** as comprised of 10 such dialogues. At the end of each learning step, the system is tested using the test set (100 test instances).

This process is repeated 20 times, i.e. for 20 rounds/folds, each time with a different, random 500-100 split, thus resulting in 20 data-points for cost and accuracy after every learning step. The values reported below, including those on the plots in Fig. 6a, 6b and 6c, correspond to averages across the 20 folds.

4.3 Learning an Adaptive Policy for a Dynamic Confidence Threshold

In the experiment presented above, the learning agent’s positive confidence threshold was held constant, at 0.9. However, since the confidence threshold itself becomes more reliable as the agent is exposed to more training instances, we further hypothesised that a threshold that changes dynamically over time should lead to a better trade-off between classification accuracy and cost for the tutor, i.e. a better *Overall Performance Ratio* (see above). For example, lower positive thresholds may be more appropriate at the later stages of training when the agent is already performing well with attribute classifiers which are more reliable. This leads to different dialogue behaviours, as the learner takes different decisions as it encounters more training examples.

To test this hypothesis we further trained and evaluated an adaptive policy that adjusts the learning agent’s confidence threshold as it interacts with the tutor (in the +UC conditions only). This optimization used a Markov Decision Process (MDP) model and Reinforcement Learning⁴, where: (1) the **state space** was determined by vari-

⁴A reviewer points out that one can handle uncertainty in a more principled way, possibly with better results, using POMDPs. Another reviewer points out that the policy learned is only adapting the confidence threshold, and not the other conditions (uncertainty, initiative, context-dependency). We point out that we are addressing both of these limitations in work in progress, where we feed each classifier’s outputted confidence level as a continuous feature in a (continuous space) MDP for *full dialogue control*.

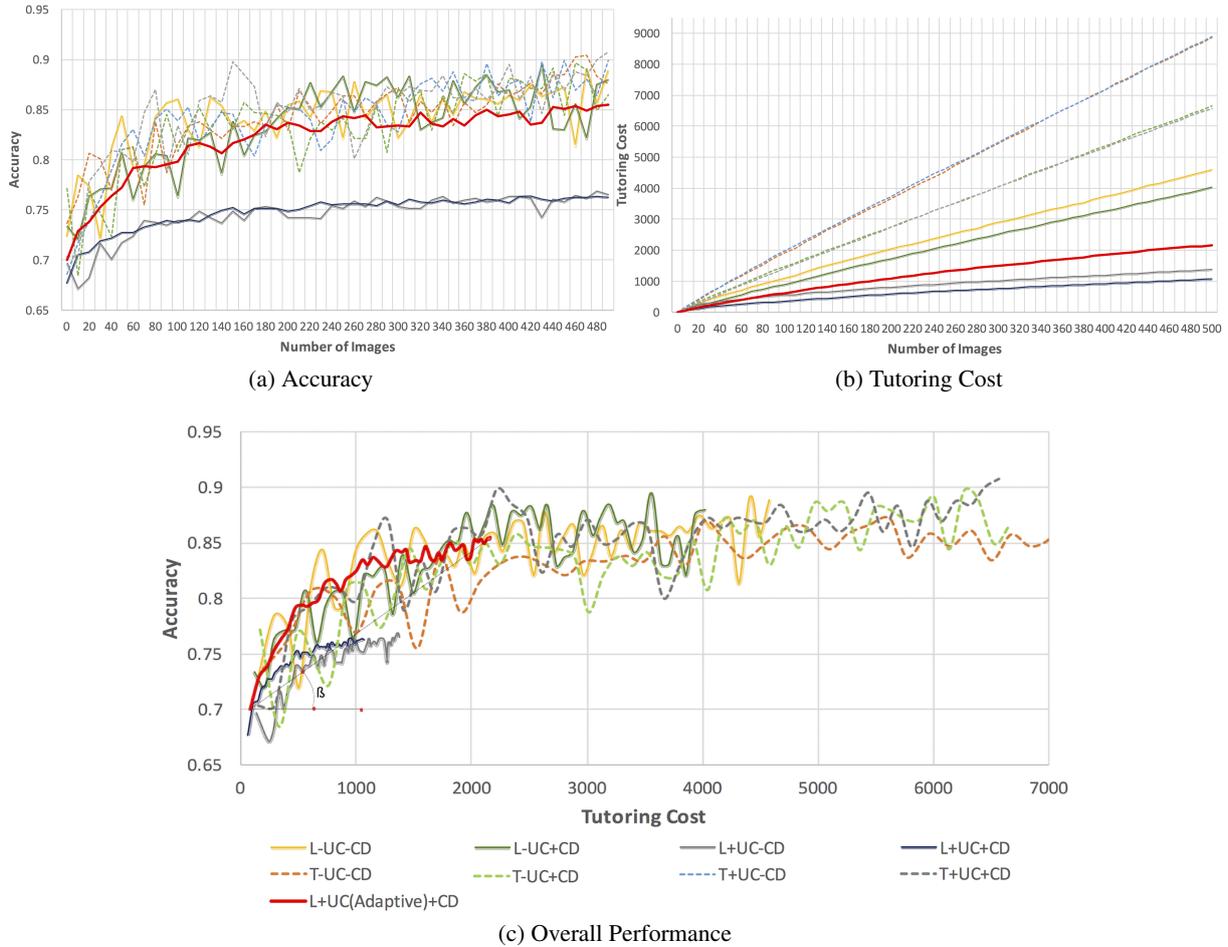


Figure 6: Evolution of Learning Performance

ables for the number of training instances seen so far, and the agent’s current confidence threshold (2) the **actions** were either to increase or decrease the confidence threshold by 0.05, or keep it the same; (3) the **local reward signal** was directly proportional to the agent’s *Overall Performance Ratio* over the previous Learning Step (10 training instances, see above); and (4) the **SARSA algorithm** (Sutton and Barto, 1998) was chosen for learning, with each **episode** defined as a complete run through the 500 training instances.

5 Results

Fig. 5 shows example interactions between the learner and the tutor in some of the experimental conditions. Note how the system is able to deal with (parse and generate) utterance continuations as in $T+UC+CD$, short answers as in $L+UC+CD$, and polar answers as in $T + UC + CD$.

Fig. 6a and 6b plot the progression of average Accuracy and (cumulative) Tutoring Cost for each of the 8 conditions in our main experiment,

as the system interacts over time with the tutor about each of the 500 training instances. The ninth curve in red ($L+UC(Adaptive)+CD$) shows the same for the learning agent with a dynamic confidence threshold using the policy trained using Reinforcement Learning (section 4.3) - the latter is only compared below to the dark blue curve ($L+UC+CD$). As noted in passing, the vertical axes in these graphs are based on averages across the 20 folds - recall that for Accuracy the system was tested, in each fold, at every learning step, i.e. after every 10 training instances.

Fig. 6c, on the other hand, plots Accuracy against Tutoring Cost directly. Note that it is to be expected that the curves should not terminate in the same place on the x-axis since the different conditions incur different total costs for the tutor across the 500 training instances. The gradient of this curve corresponds to *increase in Accuracy per unit of the Tutoring Cost*. It is the gradient of the line drawn from the beginning to the end of each curve ($\tan(\beta)$ on Fig. 4c) that constitutes our main

evaluation measure of the system’s overall performance in each condition, and it is this measure for which we report statistical significance results:

A between-subjects Analysis of Variance (ANOVA) shows significant main effects of Initiative ($p < 0.01$; $F = 448.33$), Uncertainty ($p < 0.01$; $F = 206.06$) and Context-Dependency ($p < 0.05$; $F = 4.31$) on the system’s overall performance. There is also a significant Initiative×Uncertainty interaction ($p < 0.01$; $F = 194.31$).

Keeping all other conditions constant (L+UC+CD), there is also a significant main effect of Confidence Threshold type (Constant vs. Adaptive) on the same measure ($p < 0.01$; $F = 206.06$). The mean gradient of the red, adaptive curve is actually slightly lower than its constant-threshold counter-part blue curve - discussed below.

6 Discussion

Tutoring Cost As can be seen on Fig. 6b, the cumulative cost for the tutor progresses more slowly when the learner has initiative (L) and takes its confidence into account in its behaviour (+UC) - the grey, blue, and red curves. This is so because *a form of active learning* is taking place: the learner only asks a question about an attribute if it isn’t confident enough already about that attribute. This also explains the slight decrease in the gradients of the curves as the agent is exposed to more and more training instances: its subjective confidence about its own predictions increases over time, and thus there is progressively less need for tutoring.

Accuracy On the other hand, the L+UC curves (grey and blue) on Fig. 6a show the slowest increase in accuracy and flatten out at about 0.76. This is because the agent’s confidence score in the beginning is unreliable as the agent has only seen a few training instances: in many cases it doesn’t query the tutor or have any interaction whatsoever with it and so there are informative examples that it doesn’t get exposed to. In contrast to this, the L+UC(adaptive)+CD curve (red) achieves much better accuracy.

Comparing the gradients of the curves on Fig. 6c shows that the overall performance of the agent on the gradient measure is significantly better than others in the L+UC conditions (recall the significant Initiative × Uncertainty interaction). However, while the agent with an **adaptive** thresh-

old (red/L+UC(adaptive)+CD) achieves slightly lower overall gradient than its constant threshold counter-part (blue/L+UC+CD), it achieves much higher Accuracy overall, and does this much faster in the first 1000 units of cost (roughly the total cost in L+UC+CD condition). We therefore conclude that the adaptive policy is more desirable. Finally, the significant main effect of **Context-Dependency** on the overall performance is explained by the fact in the +CD conditions, the agent is able to process context-dependent and incrementally constructed turns, leading to less repetition, shorter dialogues, and therefore better overall performance.

7 Conclusion and Future work

We have presented a multi-modal dialogue system that learns grounded word meanings from a human tutor, incrementally, over time, and employs a dynamic dialogue policy (optimised using Reinforcement Learning). The system integrates a semantic grammar for dialogue (DS), and a logical theory of types (TTR), with a set of visual classifiers in which the TTR semantic representations are grounded. We used this implemented system to study the effect of different dialogue policies and capabilities on the overall performance of a learning agent - a combined measure of accuracy and cost. The results show that in order to maximise its performance, the agent needs to take initiative in the dialogues, take into account its changing confidence about its predictions, and be able to process natural, human-like dialogue.

Ongoing work further uses Reinforcement Learning to learn complete, incremental dialogue policies, i.e. which choose system output at the lexical level (Eshghi and Lemon, 2014). To deal with uncertainty this system takes all the classifiers’ outputted confidence levels directly as features in a continuous space MDP.

Acknowledgements

This research is supported by the EPSRC, under grant number EP/M01553X/1 (BABBLE project⁵), and by the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 688147 (MuMMER project⁶).

⁵<https://sites.google.com/site/hwinteractionlab/babble>

⁶<http://mummer-project.eu/>

References

- Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *J. Artif. Intell. Res. (JAIR)*, 49(1–47).
- Ronnie Cann, Ruth Kempson, and Lutz Marten. 2005. *The Dynamics of Language*. Elsevier, Oxford.
- Robin Cooper. 2005. Records and record types in semantic theory. *Journal of Logic and Computation*, 15(2):99–112.
- Simon Dobnik, Robin Cooper, and Staffan Larsson. 2012. Modelling language, action, and perception in type theory with records. In *Proceedings of the 7th International Workshop on Constraint Solving and Language Processing (CSLP’12)*, pages 51–63.
- Arash Eshghi and Oliver Lemon. 2014. How domain-general can we be? learning incremental dialogue systems without dialogue acts. In *Proceedings of SemDial*.
- Arash Eshghi, Julian Hough, Matthew Purver, Ruth Kempson, and Eleni Gregoromichelaki. 2012. Conversational interactions: Capturing dialogue dynamics. In S. Larsson and L. Borin, editors, *From Quantification to Conversation: Festschrift for Robin Cooper on the occasion of his 65th birthday*, volume 19 of *Tributes*, pages 325–349. College Publications, London.
- A. Eshghi, C. Howes, E. Gregoromichelaki, J. Hough, and M. Purver. 2015. Feedback in conversation as incremental semantic update. In *Proceedings of the 11th International Conference on Computational Semantics (IWCS 2015)*, London, UK. Association for Computational Linguistics.
- Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Shen Furo and Osamu Hasegawa. 2006. An incremental network for on-line unsupervised classification and topology learning. *Neural Networks*, 19(1):90–106.
- Jonathan Ginzburg. 2012. *The Interactive Stance: Meaning for Conversation*. Oxford University Press.
- Julian Hough and Matthew Purver. 2014. Probabilistic type theory for incremental dialogue processing. In *Proceedings of the EACL 2014 Workshop on Type Theory and Natural Language Semantics (TTNLS)*, pages 80–88, Gothenburg, Sweden, April. Association for Computational Linguistics.
- Andrej Karpathy and Li Fei-Fei. 2014. Deep visual-semantic alignments for generating image descriptions. *CoRR*, abs/1412.2306.
- Andrej Karpathy and Fei-Fei Li. 2015. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7–12, 2015*, pages 3128–3137.
- Ruth Kempson, Wilfried Meyer-Viol, and Dov Gabbay. 2001. *Dynamic Syntax: The Flow of Language Understanding*. Blackwell.
- Casey Kennington and David Schlangen. 2015. Simple learning and compositional application of perceptually grounded word meanings for incremental reference resolution. In *Proceedings of the Conference for the Association for Computational Linguistics (ACL-IJCNLP)*. Association for Computational Linguistics.
- Thomas Kollar, Jayant Krishnamurthy, and Grant Strimel. 2013. Toward interactive grounded language acquisition. In *Robotics: Science and Systems*.
- Xiangnan Kong, Michael K. Ng, and Zhi-Hua Zhou. 2013. Transductive multilabel learning via label set propagation. *IEEE Trans. Knowl. Data Eng.*, 25(3):704–719.
- Matej Kristan and Ales Leonardis. 2014. Online discriminative kernel density estimator with gaussian kernels. *IEEE Trans. Cybernetics*, 44(3):355–365.
- Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. 2014. Attribute-based classification for zero-shot visual object categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(3):453–465.
- Staffan Larsson. 2013. Formal semantics for perceptual classification. *Journal of logic and computation*.
- Cynthia Matuszek, Liefeng Bo, Luke Zettlemoyer, and Dieter Fox. 2014. Learning from unscripted deictic gesture and language for human-robot interactions. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27–31, 2014, Québec City, Québec, Canada.*, pages 2556–2563.
- Deb Roy. 2002. A trainable visually-grounded spoken language generation system. In *Proceedings of the International Conference of Spoken Language Processing*.
- Carina Silberer and Mirella Lapata. 2014. Learning grounded meaning representations with autoencoders. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 721–732, Baltimore, Maryland, June. Association for Computational Linguistics.
- Danijel Skocaj, Matej Kristan, Alen Vrecko, Marko Mahnic, Miroslav Janíček, Geert-Jan M. Kruijff, Marc Hanheide, Nick Hawes, Thomas Keller, Michael Zillich, and Kai Zhou. 2011. A system for interactive learning in dialogue with a tutor. In *2011 IEEE/RSJ International Conference on Intelligent*

Robots and Systems, IROS 2011, San Francisco, CA, USA, September 25-30, 2011, pages 3387–3394.

- Danijel Skočaj, Matej Kristan, and Aleš Leonardis. 2009. Formalization of different learning strategies in a continuous learning framework. In *Proceedings of the Ninth International Conference on Epigenetic Robotics; Modeling Cognitive Development in Robotic Systems*, pages 153–160. Lund University Cognitive Studies.
- Richard Socher, Milind Ganjoo, Christopher D. Manning, and Andrew Y. Ng. 2013. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems*, pages 935–943, Lake Tahoe, Nevada, USA.
- Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. 2014. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218.
- Yuyin Sun, Liefeng Bo, and Dieter Fox. 2013. Attribute based object identification. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2096–2103. IEEE.
- Richard S. Sutton and Andrew G. Barto. 1998. *Reinforcement Learning: an Introduction*. MIT Press.
- Stefanie Tellex, Pratiksha Thaker, Joshua Mason Joseph, and Nicholas Roy. 2014. Learning perceptually grounded word meanings from unaligned parallel data. *Machine Learning*, 94(2):151–167.
- Yanchao Yu, Arash Eshghi, and Oliver Lemon. 2015a. Comparing attribute classifiers for interactive language grounding. In *Proceedings of the Fourth Workshop on Vision and Language*, pages 60–69, Lisbon, Portugal, September. Association for Computational Linguistics.
- Yanchao Yu, Oliver Lemon, and Arash Eshghi. 2015b. Interactive learning through dialogue for multimodal language grounding. In *SemDial 2015, Proceedings of the 19th Workshop on the Semantics and Pragmatics of Dialogue, Gothenburg, Sweden, August 24-26 2015*, pages 214–215.
- Tong Zhang. 2004. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the twenty-first international conference on Machine learning*, page 116. ACM.
- Jun Zheng, Furao Shen, Hongjun Fan, and Jinxi Zhao. 2013. An online incremental learning support vector machine for large-scale data. *Neural Computing and Applications*, 22(5):1023–1035.