Comparing attribute classifiers for interactive language grounding

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 address the problem of interactively learning perceptually grounded word meanings in a multimodal dialogue system. We design a semantic and visual processing system to support this and illustrate how they can be integrated. We then focus on comparing the performance (Precision, Recall, F1, AUC) of three state-of-the-art attribute classifiers for the purpose of interactive language grounding (MLKNN, DAP, and SVMs), on the aPascal-aYahoo datasets. In prior work, results were presented for *object* classification using these methods for attribute labelling, whereas we focus on their performance for attribute labelling itself. We find that while these methods can perform well for some of the attributes (e.g. *head, ears, furry*) none of these models has good performance over the whole attribute set, and none supports incremental learning. This leads us to suggest directions for future work.

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. smart phones, robots, and other automated systems), as well as for image search/retrieval 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 al., 2014; Socher et al., 2014). Another strand of work has focused on learning to generate object descriptions and object classification based on low level concepts/features (such as colour, shape and material), enabling systems to identify and describe novel, unseen images (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 – rather than abstract coloured shapes as in some previous work e.g. (Roy, 2002). For example, we aim to build multimodal interfaces for Human-Robot Interaction which can learn object descriptions and references in interaction with humans. In contrast to recent work on image description using 'deep learning' methods, this setting means that the system must be *trainable from little data, compositional, able to handle dialogue, and adaptive* – for instance so that it can learn visual concepts suitable for specific tasks/domains, and even new idiosyncratic language usage for particular users.

However, most of the existing systems for image description rely on training data of both high quantity and high quality 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 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 color of a "mug" or to request to be presented with more "red" things to improve its performance on the concept (see e.g. Figure 1). Furthermore, such systems could allow for meaning negotiation in the form of clarification interactions

Proceedings of the 2015 Workshop on Vision and Language (VL'15), pages 60–69,
Lisbon, Portugal, 18 September 2015. ©2015 Association for Computational Linguistics.

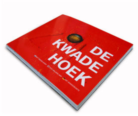
Dialogue	Image	Final semantics
S: Is this a green mug? T: No it's red S: Thanks.		$\begin{bmatrix} x_{=o1} & : & e \\ p2 & : & red(x) \\ p3 & : & mug(x) \end{bmatrix}$
T: What can you see? S: something red. What is it? T: A book. S: Thanks.		$\begin{bmatrix} x1_{=o2} & : & e \\ p & : & book(x1) \\ p1 & : & red(x1) \\ p2 & : & see(sys, x1) \end{bmatrix}$

Figure 1: Example dialogues & resulting semantic representations

with the tutor.

This paper presents initial work in a larger programme of research with the aim of developing dialogue systems that learn (visual) concepts – word meanings – through situated dialogue with a human tutor. Specifically, we compare several existing state-of-the-art classifiers with regard to their suitability for interactive language grounding tasks. We compare the performance of MLKNN (Zhang and Zhou, 2007), DAP (zero-shot learning (Lampert et al., 2014)), and SVMs (Farhadi et al., 2010) on the image datasets aPascal (for training) and aYahoo (testing) – see section 4. To our knowledge, this paper is the first to compare these attribute classifiers in terms of their suitability for interactive language grounding.

Our other contribution is to integrate an incremental semantic grammar suited to dialogue processing – DS-TTR[1] (Purver et al., 2011; Eshghi et al., 2012), see section 3 – with visual classification algorithms that provide perceptual grounding for the basic semantic atoms in the representations produced by the parser through the course of a dialogue (see Fig. 1). In effect, the dialogue with the tutor continuously provides semantic information about objects in the scene which is then fed to an online classifier in the form of training instances. Conversely, the system can utilise the grammar and its existing knowledge about the world, encoded in its classifiers, to make reference to and formulate questions about the different attributes of an object identified in the scene.

2 Related work

There has recently been a lot of research into learning to classify and describe images/objects. Some approaches attempt to ground meaning of words/phrases/sentences in images/objects by mapping these modalities into the same vector space (Karpathy and Fei-Fei, 2014; Silberer and Lapata, 2014; Kiros et al., 2014), or using distributional semantic models that build distributional representations with the conjunction of textual and visual information (Bruni et al., 2014). Other approaches, such as (Socher et al., 2014), propose Neural Network models based on Dependency Trees (DT), which project all words in a sentence into a DT structured representation to explore parents of each node and correlations between nodes.

In contrast to these approaches, which do not support NL dialogues, some approaches are designed based on logical semantic representations and some of them are incorporated with spoken dialogue systems (Skocaj et al., 2011; Matuszek et al., 2012; Kollar et al., 2013). A well-known logical semantic parser is the Combinatory Categorial Grammar (CCG) parser, which represents natural language sentences from human tutors in the logical forms. The "Logical Semantics with Perception" (LSP) framework by Kollar et al. (Krishnamurthy and Kollar, 2013) and the joint language/perception model by Matuszek et al. (Matuszek et al., 2012) are based on a CCG parser or using a CCG lexicon respectively. Although a CCG parser could generate similar logical representations to the DS-TTR parser/generator we use here, we believe that DS-TTR would show better performance than CCG in terms of handling the inherent incremental, fragmentary and highly context-dependent nature of dialogue.

The "Describer" system (Roy, 2002) learns to generate image descriptions, but it works at the level of word sequences rather than logical seman-

[1] Downloadable from `http://dylan.sourceforge.net`

tics, and uses only synthetically generated scenes rather than real images and image processing. Our approach extends (Dobnik et al., 2012) in integrating vision and language within a single formal system: Type Theory with Records (TTR). This combination will allow complex multi-turn dialogues for language grounding with deep NL semantics, including natural correction and clarification subdialogues (e.g. "No this isn't red, it's green.").

2.1 Attribute classification

Regarding attribute-based classification or description, Farhadi et al. (Farhadi et al., 2009) have successfully described objects with attributes by sharing appearance attributes across object categories. Silberer and Lapata (Silberer and Lapata, 2014) extend Farhadi et al.'s work to predict attributes using L2-loss linear SVMs and to learn the associations between visual attributes and particular words using Auto-encoders. Sun et al. (Sun et al., 2013) also build an attribute-based identification model based on hierarchical sparse coding with a K-SVD algorithm, which recognizes each attribute type using multinomial logistic regression. However, as these models require a large mass of training data, an increasing amount of research attempts to learn novel objects using 'one-shot' (Li et al., 2006; Krause et al., 2014) or 'zero-shot' learning algorithms (Li et al., 2007; Lampert et al., 2014). They enable a system to classify unseen objects with fewer or no examples by sharing *attributes* between known and unknown objects. Note that these methods ultimately focus on object class labels, using attributes as intermediate representations.

On the other hand, to learn attribute-based objects through NL interaction, some approaches learn unknown objects or attributes with online incremental learning algorithms (Li et al., 2007; Kankuekul et al., 2012). The "George" system (Skocaj et al., 2011), which is similar in spirit to our work, learns object attributes from a human tutor and creates specific questions to request information to fill detected knowledge gaps. However, the George system only learns about 2 shapes and 8 colours. Our goal is to couple attribute classifiers with much wider coverage to the formal semantics of a full Natural Language dialogue system.

3 System Architecture

We are developing 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 and classification; and a dialogue system module using DS-TTR (see below). Visual feature representations are built based on base features akin to (Farhadi et al., 2009). We do not yet have a fully integrated dialogue system, so for our experiments presented below, we assume access to logical semantic representations, that will be output by the DS-TTR parser/generator as a result of processing dialogues with a human tutor (more on this below) – and interface these representations with attribute-based image classifiers. Below we describe these components individually and then explain how they interact.

3.1 Attribute-based Classifiers used

In this research, in order to explore the best solution for attribute classification for an interactive system, we compare several methods which have previously shown good performance on image-labelling tasks – a multi-label classification model, a zero-shot learning model, and a linear SVM:

(a) MLkNN (Zhang and Zhou, 2007) is a supervised multi-label learning model based on the k-Nearest Neighbour algorithm, which predicts a label set for unknown instances. It has previously been used for scene labelling with 5 labels (sunset, desert, mountains, sea, trees) and reached a Precision of 0.8;

(b) L2-loss Linear SVM as used by (Farhadi et al., 2009). We used the published feature extraction and attribute training code[2], though we appear to have achieved slightly worse AUC results than achieved in (Farhadi et al., 2009) (see section 4);

(c) Direct Attribute Prediction (DAP) (Lampert et al., 2014), is a kind of zero-shot learning model, which implements a multi-layer classifier - the layer of attributes and the layer of labels - which apply the attribute variables in the attribute layer to decompose the object images in the label layer. This model allows the use of any supervised classification models for learning per-attribute coefficients. Once the image-attribute parameters are predicted, DAP can explore the class-

[2] From `http://vision.cs.uiuc.edu/attributes/`

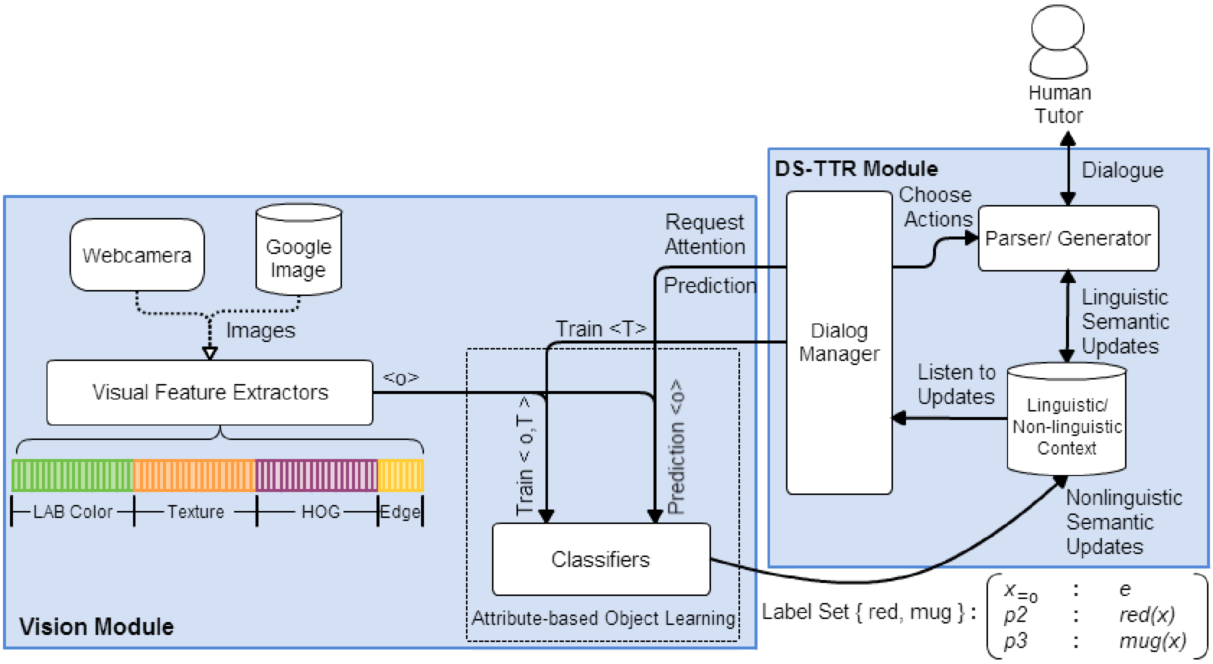


Figure 2: Architecture of the simulated teachable system

attribute relations and infer the corresponding object classes using a probabilistic model. In this paper, we reimplement the DAP zero-shot learning model based on Lampert's work; but since we are here concerned only with attribute classification we only test the first tier of their algorithm for attribute classification. (Note that although both (Farhadi et al., 2009) and (Lampert et al., 2014) implement a SVM classifier for each attribute, DAP learns the supervised model with the linearly combined χ^2-kernels rather than the original visual representations.) Note that the implementation of the DAP model is not identical to that of (Lampert et al., 2014), so our results are not directly comparable to that paper. We used the Libsvm 3.0 library (Chang and Lin, 2011), differing from the Shogun library in the original implementation for learning visual classifiers. To more directly compare the DAP model with other methods, we moreover generated the visual representation using the feature extraction algorithms by (Farhadi et al., 2009) instead of the original methods.

All models will output attribute-based label sets for novel unseen images by predicting binary label vectors. We build visual representations and binary label vectors as inputs to train new classifiers for learning attributes, as explained in the following subsections.

3.1.1 Visual Feature Representation

Following the feature extraction methods proposed by (Farhadi et al., 2009), we extract a feature representation consisting of the base features for learning to classify and describe novel objects, i.e. the colour space for colour attributes, texture for materials, visual words for object components, as well as edges for shapes.

Colour descriptors, consisting of L*A*B colour space values, are extracted for each pixel and then are quantized to the nearest 128 k-means centres. These descriptors inside the bounding box are binned into individual histograms. Edges and their orientations are detected using a MATLAB canny edge detector, which contributes to finding both edges and boundaries of objects within an image. Detected edges are quantized into 8 unsigned bins. A texture descriptor is computed for each pixel and then quantized to the nearest 256 k-means centres. Finally, object visual words are built in HOG descriptors using 8x8 blocks, a 4-pixel step size, and quantized into 512 k-means centres.

The feature extractor in the vision module presents a feature matrix with dimensions $w \times 9751$, where w is the number of training instances, and each training instance has a 9751-dimensional vector generated by stacking all quantized features, as shown in Figure 2.

3.1.2 Binary Label Vectors

For learning multi-attribute objects, the multi-label models require a label vector for each training instance. In the interactive system, an instance χ and its related label set $\eta \subseteq Y$ are given by the feature extractor and DS-TTR parser individually, where Y is a total collection of attribute-based labels. We suppose $\overrightarrow{l}$ is the binary label vector for χ, where its i-th component $\overrightarrow{l}(i)(i \in \eta)$ will take the value 1 if $i \in Y$ and -1 otherwise. Eventually, the system builds a binary label matrix with dimensions $w \times n$, where w is the number of instances and n is the total number of labels for all training instances. Each instance contains a full binary label vector. The label vectors and feature representations are used to learn new classifiers once novel object instances are learned incrementally from interaction.

3.2 Dynamic Syntax (DS)

The DS module 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 (Purver et al., 2011). 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 (see Eshghi et al (2015) for an account of how this can be achieved grammar-internally as a low-level semantic update process). Recent versions of DS incorporate Type Theory with Records (TTR) as the logical formalism in which meaning representations are couched (Purver et al., 2011; Eshghi et al., 2012), due to its useful properties. Here we do not introduce DS due to space limitations but proceed to introducing TTR.

3.3 Type Theory with Records

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

$$R_1 : \begin{bmatrix} l_1 & : & T_1 \\ l_{2=a} & : & T_2 \\ l_{3=p(l_2)} & : & T_3 \end{bmatrix} \quad R_2 : \begin{bmatrix} l_1 & : & T_1 \\ l_2 & : & T_{2'} \end{bmatrix} \quad R_3 : []$$

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 encode *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.

3.4 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 ‘mug’ are grounded in the set of classifiers trained so far.

Given a set of individuated objects in a scene, encoded as a TTR Record (see above), the system can utilise its existing ontology to output some maximal set of Record Types characterising these objects (see e.g. Fig. 1). Since these representations are shared by the DS-TTR module, they provide a direct interface between perceptual classification and semantic processing in dialogue: they can be used directly at any point to generate utterances, or ask questions about the objects.

On the other hand, 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 particular type, e.g. in Fig. 1 that the individuated object, $o1$ is a red mug. 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), and T, a record type. T is then converted automatically to an input format suitable for specific classifiers; e.g. the dialogues in Fig. 1 provide the following instances to our classifiers: $\langle o1, \{red, mug\} \rangle$ and $\langle o2, \{red, book\} \rangle$.

What sets our approach apart from other work is that these types are constructed/negotiated interactively, and so both the system and the tutor can contribute to a single representation (see e.g. second row of Fig. 1).

4 Experiments and Results

4.1 Datasets for Attribute-based classification

In order to compare the different classifiers with previous work (Farhadi et al., 2009), we perform our experiments on a benchmark dataset of natural object-based images with attribute annotations – the aPascal-aYahoo data set[3] – which is introduced by Farhadi et al. The aPascal-aYahoo data set has two subsets: the Pascal VOC 2008 dataset and the aYahoo dataset. The Pascal VOC 2008 dataset is created for visual object classifications and detections. The aPascal data set covers 20 attribute-labelled classes and each class contains a number of samples, ranging from 150 to 1000. The aYahoo dataset, as a supplement of the aPascal dataset, contains objects similar to aPascal, but with different correlations between attributes. The aYahoo dataset only contains 12 objects classes. Images in both aPascal and aYahoo sets are annotated with 64 binary attributes, covering shape and material as well as object components (see table 1). We use the 6340 images selected by (Farhadi et al., 2009) from the aPascal dataset for training and use the whole aYahoo dataset with 2644 images as the test set. As both aPascal and aYahoo data sets are imbalanced in the number of positive instances for each attribute, as shown in table 1, this might affect the performance of the models on attribute classification.

4.2 Experiment Setup

We test how well the different classifiers work on learning object attributes. We implemented several classification models – MLkNN, DAP, and SVMs as described in Section 3.1. Most work on attribute classification reports the Precision and Recall only for *object classes* – which are computed using the attribute labels – but we are directly interested in the performance of the attribute classifiers themselves. Thus we report Precision, Recall, and F1-Score for the attribute labels for each model. We also show the average scores across all attributes in table 2.

4.3 Results

We first plot the Precision and Recall for each attribute for the different models, as shown in figures 4 and 5. We take Precision to be 1 where the number of True Positives and False Negatives are both 0 for an attribute (otherwise it would be undefined).

Figures 4 - 7 compare the different methods for each attribute in terms for Precision, Recall, F1, and AUC (Area Under ROC Curve). The AUC scores are computed using an open library for computer vision algorithms – Vlfeat (Vedaldi and Fulkerson, 2010).

Table 2 shows the average scores for each method, computed across all of the attributes. The results show that DAP generally has better performance across all of the attributes, although each method has specific strengths and weaknesses.

5 Discussion

The results presented above show that while the models sometimes perform quite well on specific attributes, the performance over all attributes in general is rather poor. But we note that the shapes of the plots in the Precision and the Macro-F1 Figures, 4 and 6, are very similar, showing that the performance of the algorithms are correlated with external factors, certainly including the number of positive training instances, but also how distinctive (easy to detect) an attribute generally is. For example, the attribute 'Furry' with 250 training instances is performing relatively well using all three algorithms while other attributes with sim-

[3] http://vision.cs.uiuc.edu/attributes/

Attribute Label	aPascal	aYahoo	Attribute Label	aPascal	aYahoo	Attribute Label	aPascal	aYahoo
2D Boxy	207	146	3D Boxy	393	752	Round	39	179
Vert Cyl	195	334	Horiz Cyl	94	286	Occluded	1913	778
Tail	184	529	Head	1737	1157	Ear	1097	1048
Snout	237	708	Nose	995	345	Mouth	930	332
Hair	1095	216	Face	1022	392	Eye	1183	1061
Torso	1538	1024	Hand	811	364	Arm	1080	383
Leg	994	922	Foot/Shoe	604	719	Wing	114	11
Window	304	167	Row Wind	86	224	Wheel	336	64
Door	192	13	Headlight	162	36	Taillight	104	5
Side mirror	150	71	Exhaust	50	41	Handlebars	92	37
Engine	35	71	Text	84	388	Horn	4	145
Rein	32	284	Saddle	20	121	Skin	1396	161
Metal	581	739	Plastic	260	459	Wood	195	167
Cloth	1591	123	Furry	250	996	Glass	180	34
Feather	99	1	Wool	12	15	Clear	32	42
Shiny	432	527	Leather	6	85			

Table 1: The Number of Positive instances on each attribute in aPascal-aYahoo Datasets (aPascal for training set, aYahoo for testing Set, attributes with no testing instances removed)

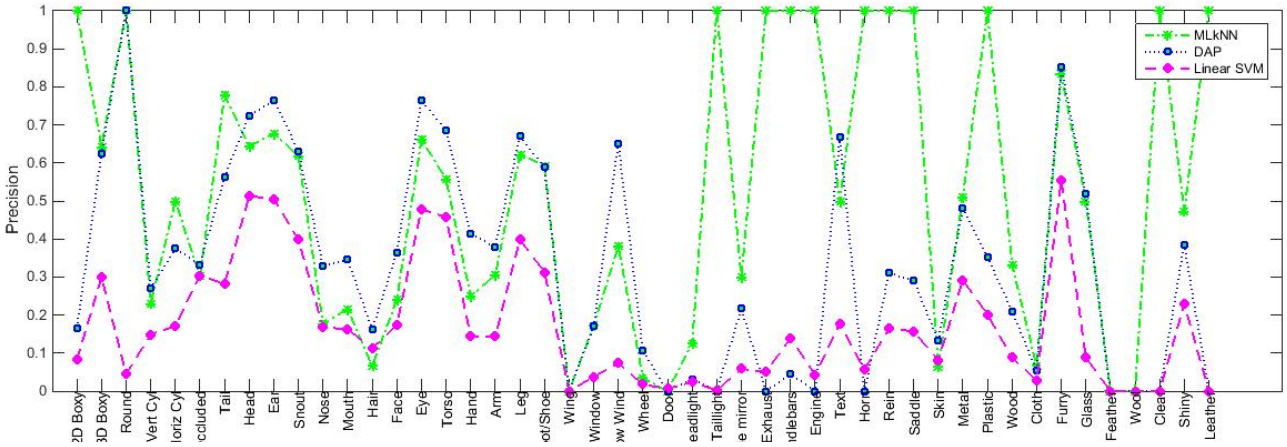


Figure 4: Precisions on each attribute for each method: MLkNN, DAP and Linear SVM (note that Precision is defined as 1 when there are in fact no True positives or False positives returned)

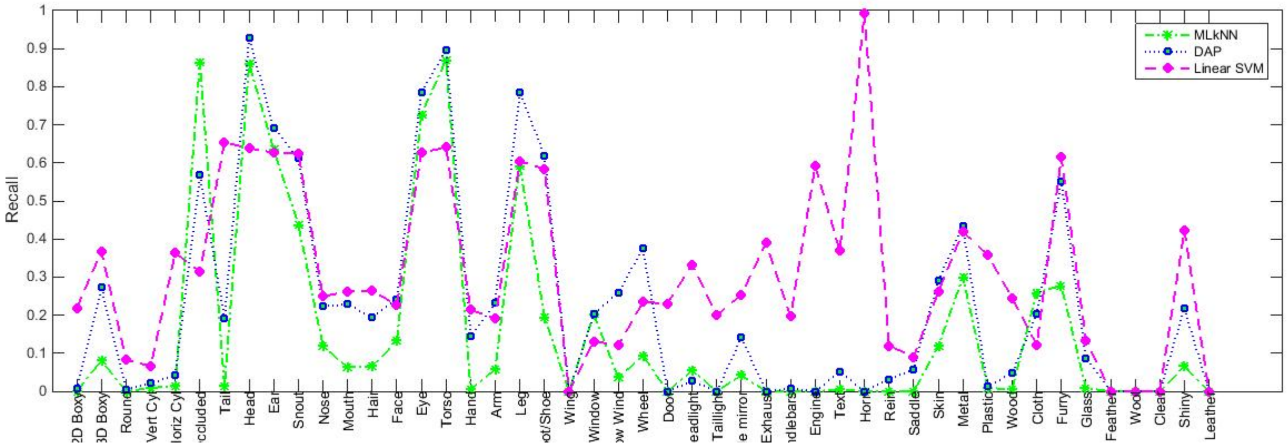


Figure 5: Recalls on each attribute for each method (MLkNN, DAP and Linear SVM)

ilar numbers of training instances are performing far worse.

Since our ultimate goal here is to create a full dialogue system that can learn concepts (word

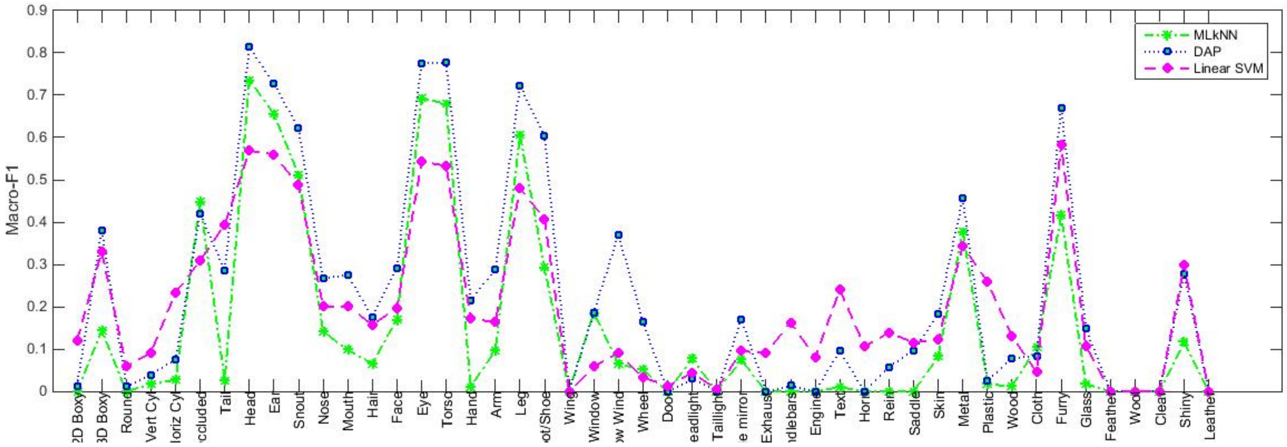


Figure 6: Macro-F1 on each attribute for each method (MLkNN, DAP and Linear SVM)

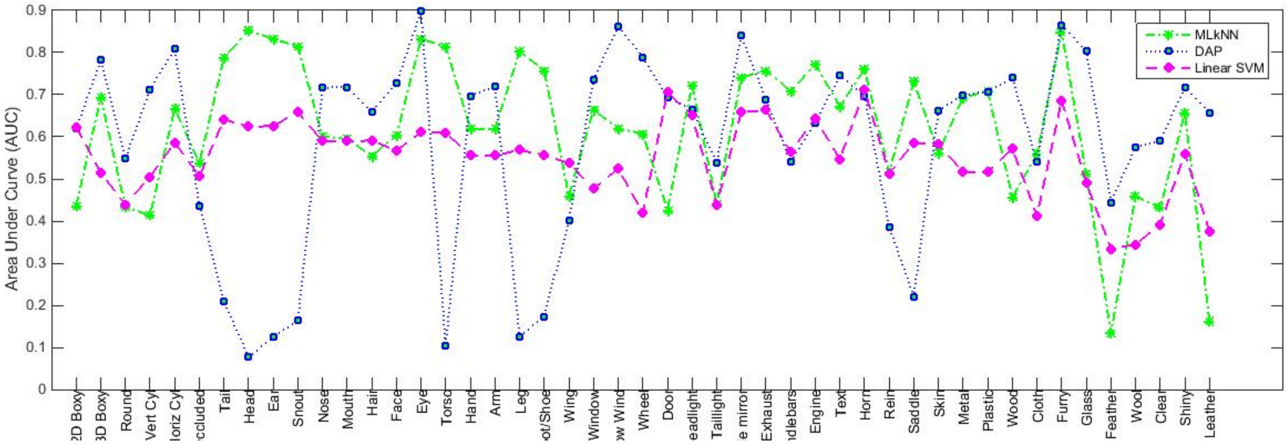


Figure 7: Area Under ROC curve for each attribute for each method (MLkNN, DAP and Linear SVM)

meanings) from human tutors, these results would lead us to pick, at least in an initial proof-of-concept system, attributes that show rapid learning rates. Presumably this is why prior work on this problem has often used 'toy' images where real image processing is not required (e.g. (Roy, 2002; Kennington et al., 2015)).

What we would need ultimately are attribute classifier learning methods which can operate effectively on small numbers of examples, and which can improve performance robustly when new examples are presented, without "unlearning" previous examples and without needing long retraining times. The dialogue abilities of the overall system will allow correction and clarification interactions to correct false positives (e.g "it's not red it's green") and other errors, and the attribute classification model must allow for such rapid retraining.

Finally we note that none of these algorithms are *incremental*. Incremental learning methods (Kankuekul et al., 2012; Tsai et al., 2014; Furao et al., 2007; Zheng et al., 2013) have been developed to train object classification networks without abandoning previously learned knowledge or destroying the old trained prototypes. These methods (such as (Kankuekul et al., 2012)) could enable systems to label known/unknown attributes gradually through NL interaction with human tutors. Incremental learning approaches can also speed up the object learning/prediction process and the system responses, rather than taking a long computational time.

We will explore these approaches in future work, to learn objects and their perceptual attributes gradually from conversational Human-Robot interaction.

Model	average Precision	average Recall	average Macro-F1
MLkNN	0.5186	0.1537	0.2372
DAP	0.3326	0.2276	0.2703
SVMs	0.1676	0.3118	0.2180

Table 2: Average scores across attribute labels for each method, trained on aPascal and tested on aYahoo

6 Conclusion

We are developing a multimodal interface to explore the effectiveness of situated dialogue with a human tutor for learning perceptually-grounded word meanings. The system integrates the semantic/contextual representations from an incremental semantic parser/generator, DS-TTR, with attribute classification models to evaluate their performance.

We compared the performance (Precision, Recall, F1, AUC) of several state-of-the-art attribute classifiers for the purpose of interactive language grounding (MLKNN, DAP, and SVMs), on the aPascal-aYahoo datasets. The results show that the models can sometimes perform quite well on specific attributes (e.g. *head, ears, torso*), but the performance over all attributes in general is rather poor. This leads us to either restrict the attributes actually used in a real system, or to explore other methods, such as incremental learning.

The immediate future direction our research will take is in developing and evaluating a fully implemented system involving classifiers incorporated with incremental learning algorithms for each visual attribute, DS-TTR, and a pro-active dialogue manager that formulates the right questions to gain information and increase accuracy.

We envisage the use of such technology in multimodal systems interacting with humans, such as robots and smart spaces.

Acknowledgements

This research is partially supported by the EPSRC, under grant number EP/M01553X/1 (BABBLE project[4]).

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.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM TIST*, 2(3):27.

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 (CSLPÄô12)*, pages 51–63.

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 Linguisitics.

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*.

Ali Farhadi, Ian Endres, and Derek Hoiem. 2010. Attribute-centric recognition for cross-category generalization. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2352–2359. IEEE.

Shen Furao, Tomotaka Ogura, and Osamu Hasegawa. 2007. An enhanced self-organizing incremental neural network for online unsupervised learning. *Neural Networks*, 20(8):893–903.

Jonathan Ginzburg. 2012. *The Interactive Stance: Meaning for Conversation*. Oxford University Press.

Pichai Kankuekul, Aram Kawewong, Sirinart Tangruamsub, and Osamu Hasegawa. 2012. Online incremental attribute-based zero-shot learning. In

[4] https://sites.google.com/site/hwinteractionlab/babble

2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012, pages 3657–3664.

Andrej Karpathy and Li Fei-Fei. 2014. Deep visual-semantic alignments for generating image descriptions. *CoRR*, abs/1412.2306.

Casey Kennington, Livia Dia, and David Schlangen. 2015. A discriminative model for perceptually-grounded incremental reference resolution. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 195–205.

Ryan Kiros, Ruslan Salakhutdinov, and Rich Zemel. 2014. Multimodal neural language models. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 595–603.

Thomas Kollar, Jayant Krishnamurthy, and Grant Strimel. 2013. Toward interactive grounded language acqusition. In *Robotics: Science and Systems*.

Evan A. Krause, Michael Zillich, Thomas Emrys Williams, and Matthias Scheutz. 2014. Learning to recognize novel objects in one shot through human-robot interactions in natural language dialogues. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, pages 2796–2802.

Jayant Krishnamurthy and Thomas Kollar. 2013. Jointly learning to parse and perceive: Connecting natural language to the physical world. *TACL*, 1:193–206.

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*.

Fei-Fei Li, Robert Fergus, and Pietro Perona. 2006. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(4):594–611.

Fei-Fei Li, Robert Fergus, and Pietro Perona. 2007. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70.

Cynthia Matuszek, Nicholas FitzGerald, Luke Zettlemoyer, Liefeng Bo, and Dieter Fox. 2012. A joint model of language and perception for grounded attribute learning. In *Proc. of the 2012 International Conference on Machine Learning*, Edinburgh, Scotland, June.

Matthew Purver, Arash Eshghi, and Julian Hough. 2011. Incremental semantic construction in a dialogue system. In J. Bos and S. Pulman, editors, *Proceedings of the 9th International Conference on Computational Semantics*, pages 365–369, Oxford, UK, January.

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 Janícek, 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.

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.

Cheng-Hao Tsai, Chieh-Yen Lin, and Chih-Jen Lin. 2014. Incremental and decremental training for linear classification. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 343–352.

Andrea Vedaldi and Brian Fulkerson. 2010. Vlfeat: an open and portable library of computer vision algorithms. In *Proceedings of the 18th International Conference on Multimedia 2010, Firenze, Italy, October 25-29, 2010*, pages 1469–1472.

Min-Ling Zhang and Zhi-Hua Zhou. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038–2048.

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.