Grounding Antonym Adjective Pairs through Interaction

Maxwell Forbes Michael Jae-Yoon Chung Maya Cakmak

Luke Zettlemoyer Rajesh P. N. Rao

{mbforbes, mjyc, mcakmak, lsz, rao}@cs.uw.edu
Department of Computer Science and Engineering
University of Washington
Seattle, WA 98102

ABSTRACT

We aim to build robots that perceive the world on a higher abstraction level than their raw sensors, and can communicate this perception to humans via natural language. The focus of this work is to enable a robot to ground antonym adjective pairs in its own sensors. We present a system where a robot is interactively trained by a user, grounding the robot's multimodal continuous sensor data in natural language symbols. This interactive training plays on the strengths of the asymmetry in human-robot interaction: not only is the training intuitive for users who understand the natural language symbols and can demonstrate the concepts to the robot, but the robot can use quick data sampling and state of the art feature extraction to accelerate the learning. Such training allows the robot to reason not only about the learned concepts, but the spaces in-between them. We show a sample interaction dialog in which a user interactively grounds antonym adjective pairs with the robot, and data showing the state of the trained model.

1. INTRODUCTION

Robots are not human. The humanoid appearance of some robots may encourage humans to consider them as human-like or to have human-like intelligence, but in reality, robots controlled by even state of the art systems must interact with humans differently than how humans interact amongst themselves.

Such interactions are asymmetric in many ways. Two of these ways are language and perception. Regarding language, humans interact in rich natural language that is often full of ambiguity and backed by assumptions about knowledge of the world, including the assumption that such knowledge is at least partially shared. Robots do not inherently have any natural language capabilities; by default, humans communicate with them through ambiguity-free program calls, often at low levels of abstraction. Regarding perception, humans have five external world senses in addition to a multitude of self-monitoring senses (such as hunger, temperature, pain, and time). Robots may have any number of sensors: cameras, pressure sensors, sonar, and more can give them lighting-fast information about their environment. However, in a way they represent different ways in which the robot can acquire numbers, rather than truly distinct senses. Thus, as in language, robots' senses operate at a lower level of abstraction than humans.

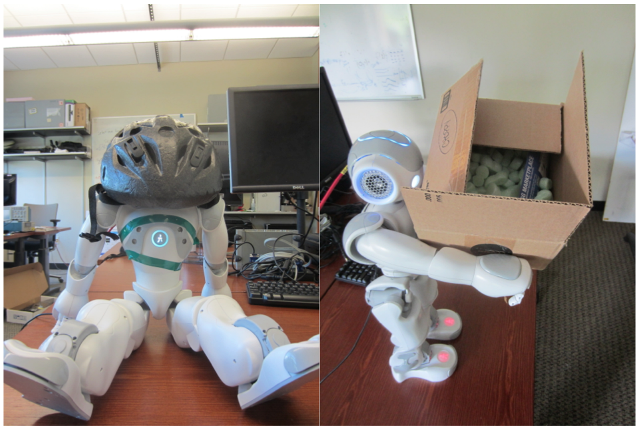

Figure 1: The NAO robot in two different configurations: "dark" (contrasted with "bright") and "heavy" (contrasted with "light").

Combining perception and language is natural to humans, but not to robots. Our work focuses on this divide. We aim to utilize the strengths of the asymmetry in human-robot interaction by combining a human's knowledge of language and the link between language and perception with a robot's multitude of sensors that provide fast access to numerical data. We do this by interactively teaching a robot about antonym adjectives, grounding them multimodally in the robot's own sensors. This raises the abstraction level with which a robot understands the link between language and perception. Doing so allows a robot more natural interactions with humans: a robot can describe its own sensors with human language. Conducting this learning interactively and online allows for a tighter feedback loop.

2. RELATED WORK

The problem we are addressing is an instance of the symbol grounding problem (SGP), first defined by Harnad in [5]. There has been a wealth of work in this area since—see the review by Coradeschi et al. in [3] for a summary of the work since 2000—even in the specific domain of multimodal symbol grounding. Needham et al. in [7] used multimodal input to autonomously learn the rules of simple tabletop games. Grollman et al. in [4] cluster multimodal sensor input in the robot's perception space, though they don't link the clusters to symbols that are semantically relevant to humans (such as language).

The work of Yu et al. in [9] is more closely related to ours and involves categorization and grounding with multimodal input, but the categorizations are unimodal nouns, and the learning is autonomous and offline rather than interactive. Chao et al. in [2] explore interactive grounding, as well as grounding at a higher abstraction level (tasks), though the inputs and grounding are unimodal. The closest work to ours is by Nakamura et al. in [6], where they use LDA to link multimodal input to multimodal noun and adjective grounding with visual, audio, and haptic dimensions. Our work is different in two ways. First, our learning is interactive and online. Second, our approach gives a robot the ability to ground the relation between adjectives as well as the adjectives themselves. Put another way, we ground the lowest abstraction level of adjective grounding, their symbol to sensor mapping, as well as one level of abstraction higher, a limited symbol to symbol relationship. This is the strength of emphasizing the asymmetry in human-robot interactions.

3. APPROACH

3.1 Inputs and Outputs

We now present an informal specification of our problem. We take as preliminary inputs pairs of natural language adjective antonyms, as well as optionally some configuration of thresholds and templates for natural language generation (NLG). Then, interactively, inputs are discrete training instances that consist of a feature vector generated from the robot's multimodal sensor space, and a single adjective from the input set. Whenever desired, the state of the grounding can be tested, and output is a report the current observations of the robot, with one phrase per antonym pair. The phrase generated can be as simple as the best adjective, or can be set to fit a language template if it is within a certain threshold (for example, if the robot is unsure, it could output "neither X nor Y").

More formally, we provide to the robot a set of n antonym adjective pairs $A = \{p_1, ..., p_n\}$, where pair $p_i = (a_1, a_2)^i$, and no adjective a^i_k is ever used twice. We also provide a set of m thresholds $\Theta = \{\theta_1, ..., \theta_m\}$, $0 \leq \theta_j \leq 1$, and matching NLG templates $T = \{t_1, ..., t_m\}$. The thresholds are used during output and will be described shortly.

We assume the robot has a set of w sensors $S = \{s_1, ..., s_w\}$ of various modalities, and that each sensor s_ℓ can be sampled to obtain a raw input vector X^ℓ of length z^ℓ, such that $X^\ell = \left[x^\ell_1, ..., x^\ell_{z^\ell}\right]$. To acquire the final set of values $\mathbf{f}_\ell = \left[\phi^\ell_1, ..., \phi^\ell_\alpha\right]$ from each sensor s_ℓ, each sensor may be transformed through a feature extraction function $f_\ell\left(x^\ell_1, ..., x^\ell_{z^\ell}\right)$. The sensor may be used raw, in which case $\mathbf{f}_\ell = X^\ell$, or f_ℓ might define a more complicated transformation. The final input vector from S is denoted $\mathbf{F} = [\mathbf{f}_1, ..., \mathbf{f}_w]$. A single training datum would consist of the pair $\left(a^i_k, \mathbf{F}\right)$, or an adjective from the user and a complete input feature vector transformed from the robot's sensors.

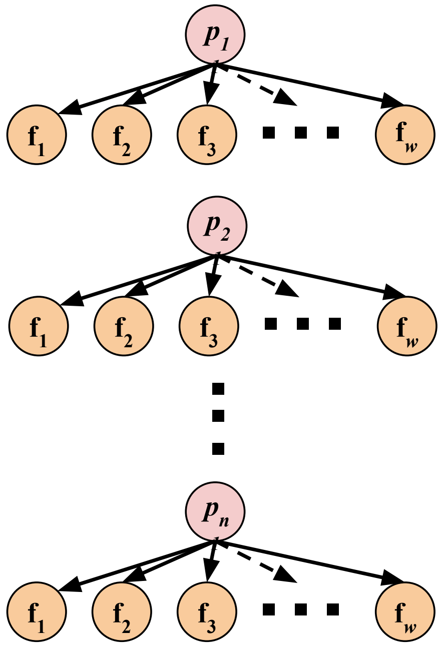

Figure 2: The graphical model used to represent the antonym adjective pairs' groundings in the multimodal sensors' feature space.

For output, the robot will sample its sensors and observe $\mathbf{F}$, and from this will produce n phrases, one for each antonym pair. Phrase i is generated from the robot's belief over p_i and natural language is constructed with the use of Θ and T. Let us denote the robot's belief over p_i given $\mathbf{F}$ as $b(a^i_k|\mathbf{F})$, where $k \in \{1, 2\}$ as adjectives are in pairs. (We will write $b(a^i_k)$ when it is clear that $\mathbf{F}$ is constant.) Each belief is the probability that the robot has observed a^i_k. As these are probabilities, naturally $0 \leq b(a^i_k) \leq 1$ and $\sum_{k=1}^{2} b(a^i_k) = 1 \ \forall i \in 1, ..., n$. Denote a^i_+ the adjective in antonym pair i with the highest $b(a^i_k)$, and a^i_- the other. The smallest θ_j such that $a^i_+ - a^i_- \leq \theta_j$ determines which language template t_j will be used. A template $t_j(a^i_+, a^i_-)$ produces a phrase such as "a^i_+" or "Neither a^i_+ nor a^i_-."

3.2 Representation

The process of grounding the antonym adjectives in $\mathbf{F}$ is the process of learning $b(\cdot|\mathbf{F})$. We model each antonym pair separately by a Gaussian naive Bayes, where the pair p_i generates all features $\mathbf{F}$. As such, we model all features as conditionally independent given the observed adjective (here $y = a^i_k$ for notational simplicity), and assume the likelihood of all features are Gaussian:

$$P(\phi^\ell_j|y) = \frac{1}{\sqrt{2\pi\sigma^2_y}} \exp\left(-\frac{\left(\phi^\ell_j - \mu_y\right)^2}{2\pi\sigma^2_y}\right)$$

A graphical model of this representation can be seen in Figure 2.

4. IMPLEMENTATION

For our system, we attempt to ground $n = 6$ antonym adjective pairs. The pairs are:

1. "heavy" vs "light"
2. "near" vs "far"
3. "bright" vs "dark"
4. "crowded" vs "lonely"
5. "uncovered" vs "touching"
6. "fast" vs "slow"

For natural language templates, we provide the system with $m = 3$ thresholds Θ and templates T. Recall that a template t_j is used for an adjective pair p_i if θ_j is the smallest in Θ such that $a^i_+ - a^i_- \leq \theta_j$. These are as follows:

1. $\theta_1 = 0.1$, $t_1 =$ "`Neither` a^i_+ `nor` a^i_-"
2. $\theta_2 = 0.3$, $t_2 =$ "`More` a^i_+ `than` a^i_-"
3. $\theta_3 = 1.0$, $t_2 =$ "a^i_+"

We use an Aldebaran NAO robot for the physical instantiation of our system. We use a total of 15 sensors for the interactive grounding, where $w = 14$ were "grounded" in that they were used as inputs to the feature extraction for learning[1]. The sensors include 1 microphone, 1 camera, 8 foot weight sensors, 3 haptic head sensors, and 2 sonar distance sensors. These sensors represent inputs from five modalities: audio, vision, weight, touch, and distance. We transform them into a final feature vector $\mathbf{F}$ of length 12 by extracting features from some sensors and compressing others.

From the camera, we extract four features from the most recent recorded frame: the brightness measurement, the number of faces detected, and the average and total movement velocities. We compress the eight foot weight sensors into three values: the of the weight on the left foot, the right foot, and combined. The two sonar sensors, located on the chest of the NAO, provide left and right sonar distances, and are used raw without transformation. Finally, the three head haptic sensors each provide binary input as to whether their region—the front, middle, and rear of the head—are being touched. We concatenate these features together to make our training feature vector $\mathbf{F}$.

To represent the Gaussian naive Bayes we use the Scikit-learn machine learning package for Python [8]. As each antonym adjective pair is represented by a separate naive Bayes, we binarize the adjective observed in each training datum.

5. DEMONSTRATION OF SYSTEM

In this section we present an example dialog between the NAO robot running our system with a human user, as well as data from the trained Gaussian naive Bayes models for different antonym adjective pairs.

5.1 Dialog

The dialog presented grounds only a subset of the available adjectives pairs in the interest of space. Specifically, "heavy" versus "light," and "bright" versus "dark" are grounded. Because each adjective pair is modeled independently, grounding further adjectives takes only additional training examples and does not interfere with grounding other pairs. Note that because of the presence of speech recognition errors, the NAO repeats each word it successfully recognizes as a confirmation.

NAO: `Please demonstrate and then say adjectives. Say 'predict' to move to the prediction phase, or 'done' to finish.`

(The user places a piece of paper on the robot's arms.)

User: *Light.*
NAO: `Light.`

(The user replaces the paper with a cardboard box filled with packing peanuts.)

User: *Heavy.* (Speech recognition error—the NAO beeps but does not repeat the word.)
User: *Heavy.*
NAO: `Heavy.`

(The user removes the box.)

User: *Light.* (User error—they intended to say "bright.")
NAO: `Light.`
User: *Bright.*
NAO: `Bright.`

(The user turns off the lights in the room.)

User: *Dark.*
NAO: `Dark.`

(The user turns the room lights back on.)

User: *Predict.*
NAO: `OK. Please configure me as desired and say 'predict' again.`
User: *Predict.*
NAO: `Light. Bright.`

(The user turns the room lights off and places the piece of paper on the NAO.)

User: *Predict.*
NAO: `Light. Dark.`

(The user replaces the paper with the box.)

User: *Predict.*
NAO: `Heavy. Dark.`
User: *Done.*
NAO: `Goodbye.`

5.2 Learned Models

Rather than learning weights like other models in machine learning, training a naive Bayes classifier involves learning averages and variances (μ, σ^2) for each class[2].

Figure 3 shows the learned model for two adjective pairs. In the left graph, we see that the brightness feature ($\mathbf{f}_6$) has the largest difference in average value between "bright" and

[1] The final sensor is the NAO's microphone array, which is used for speech recognition from the user.

[2] We assume a uniform prior over classes, though this is based on the training examples given by the user.

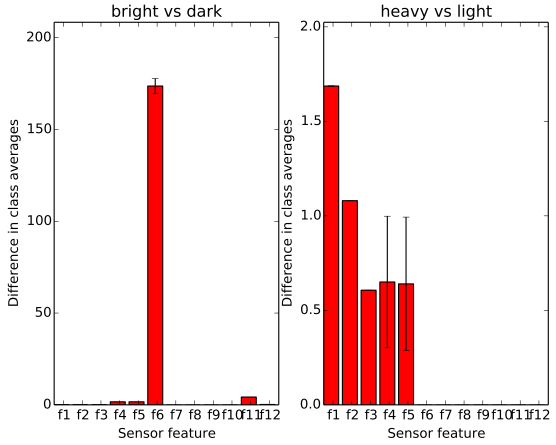

Figure 3: The differences in learned class (adjective) averages for two pairs: "bright" vs "dark" (left) and "heavy" vs "light" (right). For each pair (a, b), we have plotted the difference in the learned averages $\vec{\mu}_a - \vec{\mu}_b$ for each feature $\mathbf{f}_i$, so each bar represents one $\mu_a^{\mathbf{f}_i} - \mu_b^{\mathbf{f}_i}$. The error bars are the average variances for each pair, so $\frac{\vec{\sigma}_a^2 + \vec{\sigma}_b^2}{2}$.

"dark," so this feature will correctly play the most prominent role when determining which adjective to predict. In the right graph, for "heavy" versus "light," the first three sensors $\mathbf{f}_1$ through $\mathbf{f}_3$—which are the robot's combined total, left, and right foot sensors, respectively—correctly have a high observed difference between the two class averages. We notice that the noisy left and right sonar readings—$\mathbf{f}_4$ and $\mathbf{f}_5$—also report high differences between the class averages. However, their variances (plotted as error bars) are also quite high, so such changes will not affect correct predictions.

6. CONCLUSIONS

We present a robotic system that can ground antonym adjective pairs from continuous multimodal sensor data through interactive training with humans. Both parties benefit because the structure of our system emphasizes the asymmetries in such a human-robot interaction: the robot is granted key pieces of human knowledge—which words are antonyms and which configurations represent these adjectives—and the human can easily train a robot to describe its sensor state in natural language.

One challenge of the proposed approach is that as correlations are inherently learned, the robot may learn correlations from the particular training environment which do not correspond to the desired grounding. For example, one situation we encountered was that in order to teach the robot "touching" versus "uncovered," the user would approach the robot and put her hand on the haptic sensors located on its head. In doing so, the robot learned a correlation between "touching" and not only the positive readings of the haptic sensors, but the lower reading of the two sonar sensors, given that the user was closer to the robot when they demonstrated "touching." Discovering the relevant dimensions for the robot could be addressed by employing techniques from active learning, such as having the robot ask pointed questions (see Cakmak et al. in [1]).

An additional inherent challenge to learning adjectives is that certain adjectives have multiple meanings and so could fit in multiple antonyms groups. For example, in our own work, we encountered the fact that "light" can refer to a lack of weight, but also to the absence of darkness—a meaning for which we use the word "bright." This works in theory, but users without prior knowledge of the system do not know to follow such a distinction.

The structure of the system is such that in future work, simply increasing k from 2 to a higher number would allow the system to reason about groups of related adjectives rather than simply pairs. These words could be added interactively. We would like to explore this domain, as well as more advanced techniques that could help the user with training a larger vocabulary. For example, performing linear interpolation of relevant sensor differences between known adjectives could help the system predict to which class of adjectives a new word would belong. Furthermore, natural language processing techniques could be applied for additional signals as to how to best group adjectives.

7. REFERENCES

[1] M. Cakmak and A. L. Thomaz. Designing robot learners that ask good questions. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pages 17–24. ACM, 2012.

[2] C. Chao, M. Cakmak, and A. L. Thomaz. Towards grounding concepts for transfer in goal learning from demonstration. In *Development and Learning (ICDL), 2011 IEEE International Conference on*, volume 2, pages 1–6. IEEE, 2011.

[3] S. Coradeschi, A. Loutfi, and B. Wrede. A short review of symbol grounding in robotic and intelligent systems. *KI-Künstliche Intelligenz*, pages 1–8, 2013.

[4] D. H. Grollman, O. C. Jenkins, and F. Wood. Discovering natural kinds of robot sensory experiences in unstructured environments. *Journal of field robotics*, 23(11-12):1077–1089, 2006.

[5] S. Harnad. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1):335–346, 1990.

[6] T. Nakamura, T. Nagai, and N. Iwahashi. Grounding of word meanings in multimodal concepts using lda. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 3943–3948. IEEE, 2009.

[7] C. J. Needham, P. E. Santos, D. R. Magee, V. Devin, D. C. Hogg, and A. G. Cohn. Protocols from perceptual observations. *Artificial Intelligence*, 167(1):103–136, 2005.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011.

[9] C. Yu and D. H. Ballard. On the integration of grounding language and learning objects. In *AAAI*, volume 4, pages 488–493, 2004.