HTML Paper

SPOKEN LANGUAGE ACQUISITION VIA HUMAN-ROBOT INTERACTION

Qiong Liu, Thomas Huang, Ying Wu, Stephen Levinson

Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA

ABSTRACT

This paper presents a subproject of a challenging project that explores teaching a computer human-intelligence. In the subproject, a multisensory mobile robot is used as the interface for human-computer interaction, and spoken language is taught to the computer through natural human-robot interaction. Different from state-of-the-art speech recognizers, our approach associates speech patterns directly with sensory inputs of the robot. This approach allows our system to learn multilingual speech patterns online. Further investigation of this project will include human-computer interaction that involves more modalities, and applications that use the proposed idea to train home appliances.

1. INTRODUCTION

The idea of teaching a machine human-intelligence may be traced back to Turing's original thoughts of an intelligent machine [8]. Engineering research of an intelligent machine is useful for developing tools to assist humanity; intensive research of the intelligent machine may lead to a deep understanding of human-mind functions, human-mind evolution processes, and limitations of current computer techniques.

To approach the challenging problem of teaching a computer human-intelligence, a computer-controlled multisensory mobile robot is used in this project to interact with human when a human teaches the computer, and the spoken language acquisition research is selected as the first subproject. The terminology of "spoken language acquisition" denotes the process of a computer to learn the signal pattern and the meaning of human speech. It is also used in [2] with a similar meaning. The spoken language acquisition research has two advantages over the whole project. First, the computer's speech ability can help developers to retrieve what the computer has learned in a convenient way; this is important for debugging the system. Second, the computer's speech ability can give the system more flexibility to express itself than the robot's action ability can.

In early research in Bell Labs, Henis and Levinson built a mobile robot for spoken language acquisition [4]. Their system directly translates the speech acquired by the robot into text with a pre-trained speech recognizer. The predefined symbols (text) and grammars in the pre-trained speech recognizer strictly limit their system for natural language acquisition (e.g. acquire English and Chinese at the same time). The language acquisition approach proposed by Roy and Pentland in the late 1990s tackled the text representation problem encountered in early research by using phonemes to represent speech [6]. However, the phoneme recognition system in their design still cannot provide sufficient flexibility to the language acquisition task. To overcome problems encountered in early research, we propose using speech patterns and other robot sensory inputs directly and teaching the robot to understand human speech through natural human-robot interaction. Guided by this idea, we design an interactive and incremental learning algorithm for the robot-computer system. In our project, the algorithm uses the robot sensory inputs instead of text to explain speech signals so that the robot can learn speech in various (potentially any) languages. It also uses the speech feature representations directly for communication without translating this media (speech) into another media (text) for manipulation.

The rest of this paper is organized as follows. In Section 2, we discuss an information theory analogy of the robot training process. This discussion is useful for the robot training strategy design and our algorithm design. Section 3 focuses on the interactive and incremental learning algorithm. Section 4 reports our language acquisition experiments. The concluding remarks and future works are given in Section 5.

2. INFORMATION THEORY ANALOGY OF THE ROBOT TRAINING PROCESS

The robot training process can be considered as an information transmission process. In this process, the human teacher is considered as an information source. All sensors on the robot platform are considered as a communication channel. The robot brain (i.e. the computer) is considered as an information destination.

With this analogy, the information distribution in a sensory-data feature space is vital for the robot training and the algorithm design. Let ***x*** be a point in the feature space, ***y*** be a possible label of ***x***, and ***p(y|x)*** be the conditional probability of ***y*** at point ***x***, the information related to point ***x*** can be measured with conditional entropy

$$H(Y \mid X = x) = -\sum_{y} p(y \mid x)\log(p(y \mid x)) \,. \qquad (1)$$

As ***p(y|x)*** varies in the feature space, the information related to every feature point also varies in the feature space. If two classes $\boldsymbol{y_1}$ and $\boldsymbol{y_2}$ are separated based on the Bayes decision theory,

$$p(y_1 \mid x) = p(y_2 \mid x) \qquad (2)$$

will hold at a boundary point ***x*** of class $\boldsymbol{y_1}$ and class $\boldsymbol{y_2}$. (2) indicates that the boundary points have more information than other points. It also suggests that a point near the classification boundary is more important in terms of conveying information than a point that is distant from the classification boundary, provided that the density function varies smoothly in the feature space. Based on this fact, a robot instructor should spend most teaching effort on data that are close to a decision boundary.

2001 IEEE International Conference on Multimedia and Expo
ISBN 0-7695-1198-8/01 $17.00 © 2001 IEEE

0-7695-1198-8/01/$10.00 (C) 2001 IEEE

Moreover, the robot brain should assign sufficient resources to learn the data that are close to a decision boundary.

Based on the information theory analogy, we propose a robot training strategy as follows. When a robot responds correctly to its input, the robot teacher should consider that the robot have understood the training sample and therefore do nothing to the robot. When the robot responds incorrectly to its input, the teacher should consider that the robot does not understand the training sample and therefore should repeatedly teach the robot about the correct response to the sample. Through repeating the training process, we expect the robot to perform correct actions on the data that are similar to taught samples. This training scheme is different from the classical training scheme, in which the classifier parameters are estimated only according to the data frequency in real life and no additional trainings are done for misclassified cases. It is noteworthy that our teaching strategy aligns well with our daily experiences. For example, we always practice more on confusing words to enhance our memory even through these words may not be used as frequently as we have practiced them [10].

3. AN INTERACTIVE & INCREMENTAL LEARNING ALGORITHM

3.1 Interactive and Incremental Learning in a Static Feature Space

The main idea of our algorithm is based on Vector Quantization (VQ) and posterior probability estimation [3,7]. The VQ approach uses a set of vectors to represent data in the feature space. With this set of data, the feature space is separated into a set of regions. These regions are called Voronoi cells. The basic consideration of our approach is Voronoi cell split and deletion. Our algorithm separates an existing cell according to the task requirement, and deletes a cell when it is not used for a long time. In this way, our algorithm is not greatly affected by initialization; it does not need to maintain a large number of neighborhood connections for soft-competitive learning either. As an algorithm property, it is not difficult to prove that the error rate of our approach is asymptotically bounded as many nearest-neighbor classifiers [1].

To use representation vectors more efficiently, our algorithm uses an information related value $\boldsymbol{I_R}$ to control the cell separation. Let $\boldsymbol{x}$ be an input vector, $\boldsymbol{y}$ be a label, $\boldsymbol{R}$ be the region occupied by a Voronoi cell, $\boldsymbol{n}$ be the number of training data that fall in $\boldsymbol{R}$, and $\boldsymbol{p(y|R)}$ be the conditional probability estimation, the information related value $\boldsymbol{I_R}$ can be defined with

$$I_R = -n \cdot \sum_y p(y \mid R) \log(p(y \mid R)) \cdot \qquad (3)$$

With the estimation of $\boldsymbol{I_R}$, the algorithm can split a cell whenever the $\boldsymbol{I_R}$ of the cell is greater than a threshold δ. After the cell split, the $\boldsymbol{I_R}$ is decreased through decreasing $\boldsymbol{n}$ to a half.

Using $\boldsymbol{I_R}$ as the cell-split criterion is inspired by our information theoretical analogy of the learning process. Assuming that each sensory input is independent of other sensory input and the data falling in region $\boldsymbol{R}$ carries approximately $-\sum_y p(y \mid R) \cdot \log(p(y \mid R))$ information, the information transmitted into region $\boldsymbol{R}$ through the teaching process can be approximated by $\boldsymbol{I_R}$ in (3). When information estimation in a region exceeds a threshold, it is reasonable to believe that data falling in this region are difficult to be classified and the region requires more resources to separate the data. Our approach splits the cell that has a high $\boldsymbol{I_R}$ estimation into two to increase resources in the region. In real implementation, the threshold δ of $\boldsymbol{I_R}$ for cell separation is dynamically changed according to the available resources. In other words, this threshold is set low when the available resources are sufficient, and high otherwise. This mechanism allows a robot instructor to train a computer to distinguish minor feature differences through repeatedly training with similar examples. It aligns well with our training strategy.

The presented algorithm allocates Voronoi cells according to the task requirement, and uses Voronoi cell boundaries to approximate the optimal decision boundaries. In general, the piecewise boundary defined by the presented algorithm does not follow the direction of the underlying class boundary at a small scale. In our robot-learning algorithm, we use the learning vector quatization (LVQ) [5] approach to deal with this problem. Let $\boldsymbol{m_i}$ and $\boldsymbol{m_j}$ be the two nearest representation vectors to the input $\boldsymbol{x}$, *α and β* are learning constants. For input $\boldsymbol{x}$ with label $\boldsymbol{y}$, the LVQ boundary adjustment mechanism can be described with (4).

$$\begin{aligned} &If\,(p(y \mid R_j) > p(y \mid R_i))and(x \in R_i) \\ &m_j(t+1) = m_j(t) + \alpha(x - m_j(t)) \\ &Else \\ &m_i(t+1) = m_i(t) + \beta(x - m_i(t)) \end{aligned} \qquad (4)$$

Synthetic data experiments indicate that combing (3) and (4) allows our algorithm to use representation vectors more efficiently than using (3) only.

Compared with classical VQ based classifiers, the proposed algorithm uses the representation vectors more efficiently. It is also simpler and more efficient than classical density estimation approaches for the robot-learning task. As the proposed algorithm emphasizes the training data near decision boundary, it is somewhat related to the Support Vector Machine (SVM) techniques. But Vapnik got his idea from somewhere else [9].

3.2 Mixture of Classical Supervised and Unsupervised Learning

An ideal situation for a human to teach a robot is to give the robot correct labels for all its inputs (supervised learning). However, it is difficult to achieve this requirement in practice. In our project, we mix classical supervised learning and unsupervised learning paradigms for the robot training process. This mixture is achieved through (3), where *$p(y|R)$* is estimated with labeled data and n is estimated with both labeled data and unlabeled data. Mixing supervised and unsupervised learning paradigm together can eliminate certain requirements, such as labeling data as fast as the robot acquires, labeling a data that does not have a label, and labeling all data that a robot experiences. These requirement eliminations can greatly decrease the burden of robot teachers.

3.3 How to Handle Dynamic Signals

The approaches presented in previous subsections mainly focus on classifying static patterns. To handle dynamic signals, our algorithm represents signals with index strings and their run-lengths. We call this approach the index string approach (ISA). With this approach, an index string {15, 15, 15, 15, 2, 2, 8, 8, 8} can be coded as {15:4, 2:2, 8:3}, where the indices come from vector-quantizing short signal segments with the static approach. Let $\boldsymbol{P}$ and $\boldsymbol{Q}$ be indices, $\boldsymbol{m}$ and $\boldsymbol{n}$ be index lengths, a difference measurement $\boldsymbol{d}$ between two units {$\boldsymbol{P{:}m}$} and {$\boldsymbol{Q{:}n}$} is defined with (4).

2001 IEEE International Conference on Multimedia and Expo
ISBN 0-7695-1198-8/01 $17.00 © 2001 IEEE

0-7695-1198-8/01/$10.00 (C) 2001 IEEE

$$d = 1 \cdots if P \neq Q$$
$$d = | m - n | / \max(m,n) \cdots if P = Q \quad (4)$$

Based on the difference measurement between two units, the difference $\boldsymbol{D}$ between two index strings is defined with (5), where i is the time index of every unit in the string.

$$D = \sum_i d_i \quad (5)$$

For the robot-learning task, ISA has several advantages over classical temporal sequence processing approaches, such as the Hidden Markov Model (HMM) approach. First, the ISA can avoid impossible sequences that a HMM model cannot avoid. Second, the ISA does not allow signals that are significantly different to share one string. Third, the ISA is suitable for output sequence synthesis (e.g. instruct the robot to say learned speech). Fourth, the ISA allows the robot to perceive relatively accurate time variations in the signal. Fifth, the ISA is proper for incremental learning. Sixth, the ISA can learn input signals online. Encouraged by these advantages, we use the ISA in our robot-computer learning system.

3.4 Association Among Different Modalities

Building relations among different modalities in a robot learning system is vital for a robot to perform interesting behaviors, such as performing an action according to a speech command and naming an object with speech etc. The approach presented in previous sections can build representations for inputs from each sensor. The relation among these representations can be built based on statistics of concurrent activated representations. After relations among various representations are constructed, a machine is able to respond inputs of one modality with outputs of another modality.

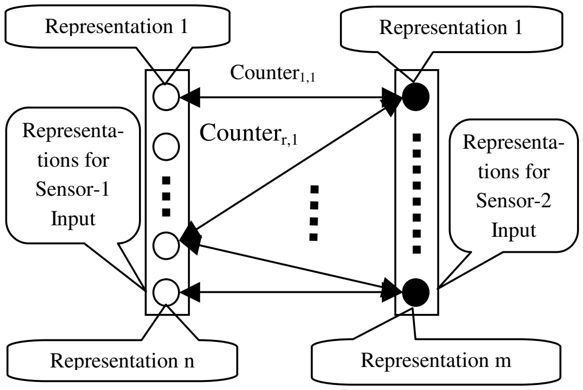

Figure 1. Representation association model between two modalities.

Fig. 1 illustrates a model for building associations among various representations. With this model, the association for every representation-pair is tracked by a dedicated counter. If representation $\boldsymbol{X_p}$ and representation $\boldsymbol{Y_q}$ happen concurrently, the $\boldsymbol{Counter_{p,q}}$ will be increased by 1. Based on the value recorded by the counter, $\boldsymbol{p(X_p,Y_q)}$, $\boldsymbol{p(X_p)}$, and $\boldsymbol{p(Y_q)}$ can be estimated. By using the Bayes equation, $\boldsymbol{p(Y_q|X_p)}$ and $\boldsymbol{p(X_p|Y_q)}$ can be calculated based on $\boldsymbol{p(X_p,Y_q)}$, $\boldsymbol{p(X_p)}$, and $\boldsymbol{p(Y_q)}$. These values are very useful for building associations between two modalities. When the representation $\boldsymbol{X_p}$ is activated by an input, the algorithm can always find the associated representation of $\boldsymbol{X_p}$ in another modality based on $\boldsymbol{p(Y|X_p)}$. On the other hand, when the representation $\boldsymbol{Y_q}$ is activated by an input, the algorithm can always find the associated representation of $\boldsymbol{Y_q}$ based on $\boldsymbol{p(X|Y_q)}$. In our current system, the associated representations are selected based on the maximum value of $\boldsymbol{p(Y|X_p)}$ or the maximum value of $\boldsymbol{p(X|Y_q)}$.

4. EXPERIMENTS

We are presently using our algorithm to enable a mobile robot to learn spoken language through its interaction with humans. The system architecture is described in Fig. 2. In this system, we use a dual CPU Octane as the "brain" of the learning system. The multimodal robot platform is used for signal acquisition through many different sensors. These sensors include a video camera, 2 microphones, 4 tactile sensors, a digital compass, a tilt sensor, a temperature sensor, bumper switches etc. The robot platform may also perform some actions through its propelling motor and steering motor. The Octane computer and the robot platform is connected through a pair of wireless modems and a pair of NTSC transmitter/receiver. Through these communication channels, the computer can request the robot to collect required signals with its multimodal sensors, or command the robot to move around according to computer decisions.

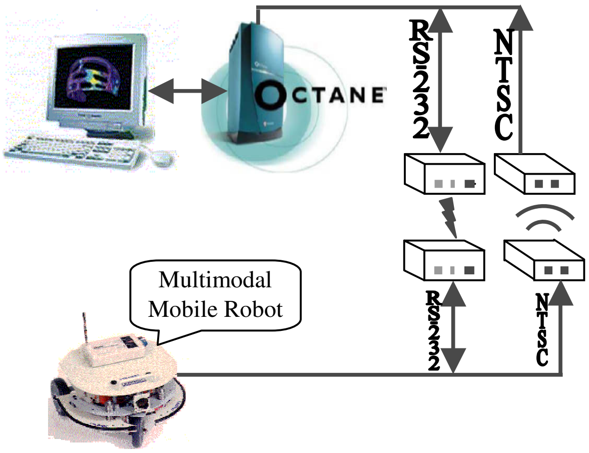

Figure 2. Multimodal Robot Learning System

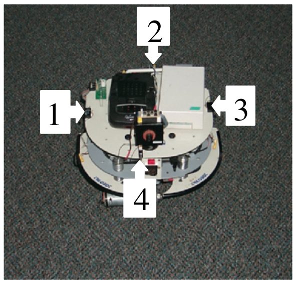

Figure 3. Touch sensors are installed on the robot chassis 1.Right sensor; 2. Back sensor; 3. Left sensor; 4. Front sensor.

In this experiment, we teach the robot to move according to our command through pushing corresponding touch sensors. More specifically, we want the robot to make associations between our commands (in any language) and our pushing forces so that we can use our language to substitute our pushing action after we teach the robot these associations. For the robot to feel

0-7695-1198-8/01/$10.00 (C) 2001 IEEE

the direction of pushing force, we added four touch sensors on the robot chassis. They are mounted on the robot as shown in Fig. 3. With these touch senesors, we can avoid damaging the robot motors by forcing them to move according to our commands.

Since the robot only has two motors to drive the platform, we can only teach the robot 6 actions. These actions are "forward", "back", "straight", "stop", "left", and "right". They are directly related to the robot's feeling on touch sensors. These relations are listed in Table 1. The relation shown in Table-1 is consistent with common sense.

Table-1 Relations between the robot action and the direction of the force

ROBOT ACTIONS	TOUCHED SENSORS
Forward	Back
Back	Front
Straight	Left and Right
Stop	Front and Back
Left	Right
Right	Left

To test the robot's ability of learning different sounds, we pronounce the corresponding commands of these actions in both English and Chinese. When we have visitors from Japan, we also ask the visitors to pronounce these commands in Japanese. The teaching strategy in the experiment follows the rules we presented in section 2.

These commands and their associated touch feelings are taught to the robot through our interactions with the robot. For example, suppose we want to teach the robot the utterances "forward" and "back". At the beginning, the robot cannot decide if there is any difference in meaning between these two sounds. So, it is possible to move forward when we say "back". This means that the feature vector of the utterance "back" locates a representation vector that has a different label, whose meaning is to instruct the robot to move forward. Our algorithm can recover from this problem in the following way. In this case, the right thing for us to do is to say "back" again to the robot, and label the audio input through touching the related sensors from time to time. If the robot cannot understand us, repeat the same word and label it again whenever possible. When we pronounce the same sound to the robot again and again, the corresponding Voronoi cell will be visit again and again, and a new cell will finally be generated in nearby regions. The representation vectors of the Voronoi cells will be changed gradually during the training. The conditional probability (related to the semantic meaning) calculated within the cells will also be updated through the training procedure. After the generation of the new cell, the computer will have more cells to represent the intensively taught data. The building of new Voronoi cells may increase the local resolution for distinguishing different inputs. After we teach the robot the semantic meaning of the new cell through touching related sensors, the robot should be able to distinguish the semantics of the utterances "forward" and "back". It is still possible for the robot to misunderstand "forward" and "back" with other sounds, but we can help clarifying these meanings through more interactions with the robot.

The speech experiments of our algorithm on the multi-sensory mobile robot have been successful. When the mobile robot starts to learn, it knows nothing about any sound as well as the meaning of any sound. Through interacting with its human teachers, the robot acquires audio information online, and relates different audio inputs to different actions according to the human-taught meanings. The robot can distinguish these commands after we teach it for about 20 minutes. Following the interactive training, when we say a command to the robot in English, Chinese, or Japanese, it can move according to our commands. The accuracy of the robot responses is about 100%. Demonstrations of this project can be found at *http://www.ifp.uiuc.edu/~q-liu2/research.html*.

5. SUMMARY AND FUTURE WORK

In this paper, we present our work on spoken language acquisition. This work is the first subproject of our autonomous learning robot project. The next step of the robot project is going to involve sound mimic experiments with the robot learning system. Providing the system with "speak back" ability will greatly increase the number of actions that a robot can perform. In the long run, we prepare to expend our learning framework to many other modalities, and enable the robot to express its sensory detection and internal state in speech and body motions. Moreover, we are also interested in applications that use the proposed framework to train home appliances.

References

1. T. M. Cover, and P. E. Hart, "Nearest Neighbor Pattern Classification," *IEEE Trans. Inform. Theory*, vol. IT-13, pp.21-27, January 1967.
2. A. L. Gorin, S. E. Levinson, and A. Sanker, "An Experiment in Spoken Language Acquisition," *IEEE Transaction on Speech and Audio Processing*, vol. 2, no. 1, part II, January, 1994.
3. R. M. Gray, "Vector Quantization," *IEEE ASSP Magazine,* 1:4-29, April 1984.
4. E. A. Henis, and S. E. Levinson, "Language as part of sensorymotor behavior," *Proc. AAAI Symposium*, Cambridge, MA, November 1995.
5. T. Kohonen, *Self-Organizing Maps*. Springer, 1995.
6. D. Roy, and A. Pentland, "Learning Words from Natural Audio-Visual Input," *Proc. Int. Conf. Spoken Language Processing*, vol. 4, pp. 1279, Sydney, Australia, December 1998.
7. D. G. Stork, R. O. Duda, and P. E. Hart, *Pattern Classification and Scene Analysis*. Unpublished book section, 1999.
8. A. M. Turing, "Computing machinery and intelligence," *Mind,* vol. 59, pp. 433-460, 1950.
9. V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer, 1995.
10. Q. Liu, S. E. Levinson, Y. Wu, T. S. Huang, Interactive and Incremental Learning via a Mixture of Supervised and Unsupervised Learning Strategies, Proceedings of the Fifth Joint Conference on Information Sciences, vol. 1, pp. 555-558. Atlantic City, NJ, U.S.A., February 27-March 3, 2000.

2001 IEEE International Conference on Multimedia and Expo
ISBN 0-7695-1198-8/01 $17.00 © 2001 IEEE

0-7695-1198-8/01/$10.00 (C) 2001 IEEE