A Sub-symbolic Process Underlying the Usage-based Acquisition of a Compositional Representation:

Results of Robotic Learning Experiments of Goal-directed Actions

Yuuya SUGITA Jun TANI
RIKEN Brain Science Institute 2-1 Hirosawa, Wako-shi, Saitama, 3510198, JAPAN
Email: {sugita, tani}@bdc.brain.riken.go.jp

***Abstract*—We propose a sub-symbolic connectionist model in which a compositional system self-organizes by learning a provided set of goal-directed actions. This approach is compatible with an idea taken from usage-based accounts of the developmental learning of language. The model explains a possible continuous process underlying the transitions from rote knowledge to systematized knowledge by drawing an analogy to the formation process of a regular geometric arrangement of points. An experiment was performed using a simulated mobile robot reaching or turning toward a colored target. By using an identical learning model, three different types of combinatorial generalization are observed depending on the provided examples. Based on the experimental results, a dynamical systems interpretation of conventional usage-based models is discussed.**

I. Introduction

This study examines the mechanisms in the human mind that are involved in the shift from unrelated rote knowledge acquired by learning examples of objects or events into a flexible conceptual system by which we can conceive something not experienced as a recombination of the examples. In related work, Skinner argued in [1] that a reusable unit emerges as a by-product of the acquisition of multiple examples containing the reusable unit. He pointed out that a minimal unit seldom appears by itself as a whole example of stimuli and responses. Tomasello reported in [2] that infants can appropriately use holophrases, which are indivisible sentences such as "lemme-see," in a communicative context before understanding the reusable units such as "let," "me" and "see."

It is, however, difficult to reveal the mechanism underlying the observation that the utilization of the whole concept precedes the acquisition of "parts", that appear to constitute the whole. In Tomasello's usage-based account of language development, the transition of the performance is explained in terms of the acquisition of a new type of smaller and more abstract symbolic repertoire, which explains a new type of combinatorial generalization. This explanation, however, simply rephrases the discontinuity of a performance as another discontinuity of internal representation, and so the questions arises as to how the new repertoire is acquired. This suggests the requirement of a continuous underlying mechanism in order to provide a longitudinal explanation throughout the transition.

This study investigates a new sub-symbolic implementation of a usage-based learning model in the domain of goal-directed actions of a simulated agent. Our model focuses on the systematic relationships among whole concepts instead of dealing with reusable conceptual parts explicitly. Each concept is embedded as points in a conceptual space that is implemented as an *n*-dimensional vector space. The geometric arrangement of these points represents the underlying combinatoriality among them. For example, a system of six actions specified by every possible combination of one of three objects and one of two operations is represented as a triangular prism, as shown in Fig. 1. Even if the positions of some actions are unknown, they can be inferred by the geometric regularity. Furthermore, this framework explains the transitions from rote knowledge to systematic knowledge in terms of a continuous internal process. The emergence of the regularity involved in the transitions can be realized by the continuous motion of each point. Moreover, a link between an action and its internal representation, namely a point corresponding to the action, does not change through the transition, whereas the conventional implementation undergoes the replacement of a holistic symbol with a combination of elemental symbols. Thus, the proposed approach provides a possible dynamic process underlying the conventional usage-based models.

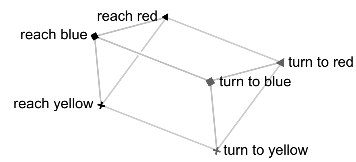

Fig. 1. Systematic relationships among concepts are represented based on the regularity of a geometric structure.

II. Experimental Setting

In our experiment, a simulated mobile agent learns an incomplete subset of 36 different goal-directed actions; the actions are characterized by combinations of a target object, an operation on the target, and an optional verb modifier. The learning is conducted in a supervised manner. The agent is designed based on a mobile robot, and has a color camera with a range of view of 120 degrees and two rotating motors driving each of its two wheels. In each experimental trial, the agent was required to perform either of two operations `reach` or `turnto` on one of six colored objects (`blue`, `cyan`, `green`,

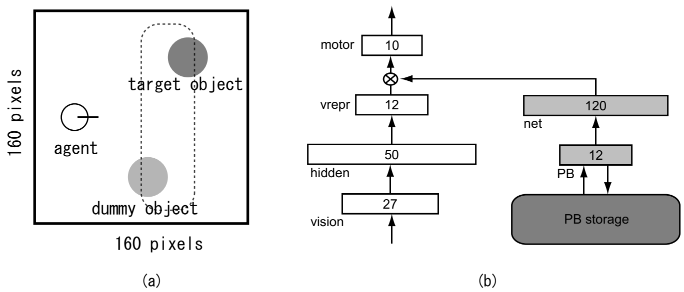

Fig. 2. (a) An experimental environment: a simulated agent performs 36 types of goal-directed actions in the environment where a target colored object and an optional dummy colored object are placed randomly within a dashed square. (b) The connectionist architecture of the model: A rectangle represents a layer. The number of nodes contained in the layer is denoted by the number in the rectangle. The gray layers have a slower time constant than do the white layers. PB (parametric bias) storage is a working area.

`yellow`, `orange`, and `magenta`) in the environment shown in Fig. 2(a), where one or two objects, one of which is the target, are randomly placed. In actions involving `reach`, the agent is required to move toward the target and then stop just before touching the target. In the `turnto` actions, the agent must pivot toward the target. The operation `turnto` takes a verb modifier, which designates the offset angle (`-30`, `-18`, `0`, `+18`, and `+30`) from the agent's center to the final position of the target in the agent's visual field. A negative angle indicates the offset to the left.In the following, the action is denoted as a concatenation of labels of one of the operations, one of the targets, and optionally one of the offset angles, for example, `turnto-blue+18`. In some situations, `turnto` with one of the angles, such as `turnto+18`, is regarded as an operation. It should be noted that the labels are used only for convenience. Any explicit information about the relationships among the actions is not presented to the agent throughout the experiment.

As mentioned above, the actions are embedded in the concept vector space through the learning process. Unlike conventional associative learning between an action and a vector, the vector is not provided *a priori*. Instead, the geometric arrangement of the vectors self-organizes the structure reflecting the relationships among the actions. The learning model that acquires this structure-preserving map between the actions and the vectors is a connectionist network, as shown in Fig. 2(b). The network consists of two parts, each of which is tailored to its own functions. One part is a base-level network (base-net), which takes the visual information from the camera as input and outputs the angular velocities of the wheels (the left side of Fig. 2(b)). The base-net is essentially a conventional layered neural network except that it has second-order connections [3] between the vrepr and motor layers. This special mechanism enables the base-net to switch its function. Depending on the action to be performed, the base-net generates different motor values for the same vision input. The second-order connection is controlled by the meta-level network (meta-net) depicted on the right side of Fig. 2(b). The meta-net is also a conventional layered network. As input, meta-net takes a vector encoding an action once at the beginning of the action; then, it outputs the weights of the second-order connections constantly until the action finishes. The input layer works just like the conventional parametric bias (PB) layer [4], which has an infinitely long time constant; therefore, we name this the PB layer. PB storage is the working area during the learning, as will be explained later in detail.

An experimental session consisted of three phases: creation of training data, learning of the data, and evaluation of the performance. The training data were created by sampling sensor-motor time series involving actions generated in an algorithmic manner. Then, the network learned a part of the data in an offline manner. Four sessions were conducted with supervised data of different sparseness, each of which contains 4, 8, 14, and 21 out of 36 actions, respectively. After the training error of the network decreased sufficiently, the performance was evaluated. A PB vector of an unseen action was computed by recognizing unused training data, as explained below. The agent, which was controlled by the network, was tested to determine whether it could perform an unfamiliar action in a novel environment by using the PB vector. In the remaining sections, each phase is explained in detail.

A. Phase 1: Generating Examples by Teaching Programs

For each of the 36 actions, 120 patterns of the action were recorded as teaching examples in different environments. In 20 out of the 120 cases, only a target object was placed in the stage, and in the remaining 100 cases, a dummy object was placed in addition to the target object. The dummy object was chosen from among five objects other than the target, and 20 patterns were sampled for each object. Both the target and optional dummy object were arranged at random positions within the area range shown by the dashed square in Fig. 2(a). Any arrangement in which the target was occluded by the dummy at the home position of the agent was omitted.

Each training example is a sensor-motor time series involving the pattern of an action generated by a manually coded teaching program. The program calculates the desired rotation speed of the two wheels of the agent from the position of a target taken from a camera image at a constant time interval in order to accomplish the specified action. Therefore, the network basically learns to mimic the program. This approach might seem inappropriate if the agent were to learn the examples by rote. However, the actual objective of the learning is to recognize the relationships among the provided exemplars in an unsupervised manner. Again, note that the abovementioned PB vectors are not under supervision.

The visual and motor information is recorded in the form of 27-dimensional and 10-dimensional vectors, respectively (see Fig. 2(b)). The visual field is composed of nine vertically divided regions. Each region is represented by the fraction of the region covered by colored patches and the dominant hue of the patches in the region. The hue is encoded by the position $(\cos\theta, \sin\theta)$ in the color circle, where pure red, yellow, green, and blue are represented as $\theta = 0°, 90°, 180°,$ and $270°$, respectively. Therefore, the visual input vector does not explicitly have the position of the target. The desired speed of the wheel takes a real value ranging from -0.2 to 1.0. A negative value indicates reverse rotation. The motor vector is composed of two five-dimensional real-valued vectors, each of which represents the speed of the wheel in the form of $[f(0), f(0.25), f(0.5), f(0.75), f(1.0)]$, where f is a Gaussian distribution with the mean of the desired speed and sigma of 0.25. This redundant encoding increases the robustness against the noise output by the network.

B. Phase 2: Batch Learning

The network learns incomplete parts of the 36 actions in a batch manner by employing the data prepared in the previous phase as the supervising signal. The learning process is formulated as a conventional iterative, steepest descent optimization with respect to the error function E, defined in (1). The model has two types of parameters to be optimized: one is the vector W consisting of all the connection weight values of the network; and the other is the set PB consisting of PB vectors pb_i for all supervised actions $i \in \mathcal{A}$.

$$E(W, PB) = \sum_{i \in \mathcal{A}} E_i(W, pb_i) \qquad (1)$$

$$E_i(W, pb_i) = \sum_{j=0}^{119} \sum_{t=0}^{l_{ij}} E_{ij}(t; W, pb_i) \qquad (2)$$

$$E_{ij}(t; W, pb_i) = \|\hat{m}_{ij}(t) - m(v_{ij}(t); W, pb_i)\|^2, \qquad (3)$$

where l_{ij} is the length of the j-th training data of action i, $\hat{m}_{ij}(t)$ is the desired motor vector corresponding to the visual vector $v_{ij}(t)$ at the time step t in the training data, and $m(v_{ij}(t); W, pb_i)$ is its actual value generated by the network under the condition that the connection weight is W, and the PB vector for the action is pb_i with the identical vision input. The parameters W and PB are updated simultaneously by learning all the provided data in a batch manner. The learning procedure is implemented by using the conventional back-propagation algorithm. At the beginning, all the connection weight values are randomized with a small value, and $pb_i, \forall i \in \mathcal{A}$ are set to the zero vector. All of the PB vectors reside in the storage because the values of the PB nodes are switched so that the network can learn all the given actions at the same time. The following procedure is then conducted 30,000 times.

(1) Do the following for each action i in $\mathcal{A}$:
   - (1.1) Load the stored pb_i to the PB nodes.
   - (1.2) For each of the 120 sensor-motor time series, calculate the delta errors of connection weight $\partial E_{ij}/\partial W\,(t; W(T), pb_i(T))$ and of PB vector $\partial E_{ij}/\partial pb_i\,(t; W(T), pb_i(T))$ by using the back-propagation algorithm.
   - (1.3) Update pb_i by using the summation of all the delta errors of pb_i for all time steps t of all time-series j of the action i, and store the updated vector in the storage.

(2) Update W by using the summation of all the delta errors of W for all time steps t of all time-series j of all the provided actions $i \in \mathcal{A}$.

Thus, the connection weights capture the common characteristics among all of the actions and play a background part, whereas each PB vector is specialized to its corresponding action. In the analysis of the experimental results, we observe

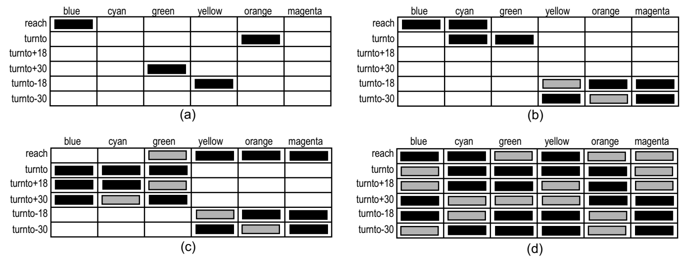

Fig. 3. The result of experiments 1, 2, 3, and 4 are shown in (a), (b), (c), and (d), respectively. A black box represents a trained action, and a gray box represents an action acquired as a recombination of the provided data.

the acquired geometric structure constructed by the PB vectors in the conceptual space.

C. Phase 3: Examining the Generalization Capability

Two aspects of the generalization capability of the agent were tested: 1) transfer of the skill to a novel environment and 2) recombination of the supervised actions into an unfamiliar action. In order to examine the transfer of skill to a novel environment, the agent was tested to determine if it could accomplish each known action in 280 novel environments, where a target and dummy object were placed in a systematic manner. The PB vectors acquired through the second phase were employed. This test reveals the kind of information kept in the vectors. If the vector codes only specific trajectories of taught examples without generalization, it is impossible to generate a goal-directed action in a different environment. In order to investigate the recombination of supervised actions into an unfamiliar action, the PB vector encoding a novel action $i' \notin \mathcal{A}$ must be examined. The vector can be computed by the *recognition* procedure. The algorithm is basically identical to the learning procedure, except that W is not updated. By employing 30 out of 120 examples of the action i' produced in the first phase, $pb_{i'}$ is optimized with regard to the error function for the action i' defined in (2) of Section II-B by using W acquired in the second phase. Once $pb_{i'}$ is obtained for each unseen action, the generation test can be conducted in the same way as in the trained action cases.

III. Results

We next observe the changes of the generalization capability depending on the sparseness of the provided examples. In Fig. 3, the degrees of generalization are compared among four experimental sessions. A trained action is indicated by a black box in the figure. In all the sessions, all the trained actions were regenerated successfully; this means the agent could accomplish the goal in more than 80 percent of the test environments explained above. A gray box shows an action achieved by the combinatorial generalization without extra teaching. The criteria of success for the novel action are identical to that for the trained action. In the remainder of this section, the results are discussed only from the viewpoint of the performance. We'll re-examine issues about the underlying mechanism in the next section.

Experiment 1 – Learning by Rote: In this case, no combinatorial generalization was observed because of very sparse training data (Fig. 3(a)). This suggests that the agent regarded the provided actions as being holistic; namely, it could not find any reusable parts such as an operation and a target.

Experiment 2 – A Local Compositional System: As training data increases, two novel actions `turnto-yellow-18` and `turnto-orange-30` were acquired without learning exemplars (Fig. 3(b)). This implies that the local compositional system is self-organized because one of the reusable operations – `turnto-18` and `turnto-30` – and one of the reusable targets – `yellow`, `orange`, and `magenta` – could be composed in any possible way, including unfamiliar combinations. However, the agent could not acquire `reach-green` and `turnto-blue`. This means that the agent failed to find the commonalities between the two operations `reach` and `turnto`.

Experiment 3 – Two Independent Compositional Systems: Two separate local compositional systems emerged when further training data were added. One is system { `turnto`, `turnto+18`, `turnto+30` } $\times$ { `blue`, `cyan`, `green` }, and the other is system { `reach`, `turnto-18`, `turnto-30` } $\times$ { `yellow`, `orange`, `magenta` }. They are independent of each other since the targets of one system cannot be applied to the operations of the other system. The result can be interpreted as the categorization of targets based on operations applicable to these targets.

Experiment 4 – Operation (Target, Offset): Finally, all the possible actions were acquired when the robot was trained with examples consisting of 21 out of the 36 actions. In addition, the agent could re-generate some actions that have intermediate offsets such as `turnto-blue-24` by recognizing newly created examples of the actions. This suggests that the similarity based on the offset values is understood. Therefore, the operations concerning `turnto` could have the structure `turnto` $\times$ `OFFSET` (= { `-30`, `-18`, `0`, `+18`, `+30` }), and the `reach` operation exists separately. Furthermore, all six targets form a class `TARGET` (= { `blue`, `cyan`, `green`, `yellow`, `orange`, `magenta` }), because they can be applied to all operations equally. Thus, it turns out that the argument structure `TARGET` $\times$ ((`turnto` $\times$ `OFFSET`) + `reach`) emerges.

IV. Analysis

For the analysis of the acquired structure in the PB space, we discuss the underlying mechanism of the combinatorial generalization proposed in Section I: the geometric regularity self-organized in the conceptual space. A primary objective of the analysis is to bridge the gap between the symbolic behavior of the system and its sub-symbolic implementation. In other words, the possible realization of functional compositionality is investigated [5].

Figure 4 shows the concept structure underlying the local compositional system observed in the second experiment. PB vectors for six actions included in the system are displayed (see also Fig. 3(b)). The displayed vectors are obtained through the learning process for trained actions and through the recognition process for untrained actions. The original 12-dimensional vectors are projected onto a two-dimensional plane computed by applying the conventional principal component analysis (PCA) method to the six vectors. The accumulated contribution rate up to the second principal component (PC) is 0.79. A regular structure similar to the prism shown in Fig. 1 is observed in the figure, although the third and subsequent PCs show irregularity. Thus, an unfamiliar action has the "correct" position in the concept space. Meanwhile, no regularity is found in the plot of actions { `reach`, `turnto` } $\times$ { `blue`, `cyan`, `green` }. This is consistent with the performance that no combinatorial generalization was realized with regard to the abovementioned actions. A similar result is found in the first experiment, where no generalization was realized.

In the third experiment, we can find two separate regular structures in accordance with the observed performance. Being consistent with the observed incompatibility between the two systems, the two structures exist on different sub-spaces. In other words, each of the systems utilizes its dedicated representation of an element and a composition mechanism.

A new facet is discovered in the concept space in the last experiment. Not only a structure representing the relationships among elements of different roles but also one representing the similarity among elements within a role are observed clearly. The former is the congruency of sub-structures, which is similar to the structure found in the second and third experiments, and the latter, with regard to each role of a target and an operation, are shown in Figs. 5(a) and (b), respectively.

The projection plane of Fig. 5(a) is chosen by applying PCA to representative vectors of the targets obtained by averaging the PB vectors for all operations for each of the targets. If a component of a target and an operation in the PB vectors are independent of each other, this method averages away the operation information. This assumption is shown to be true later in this paper. The accumulated contribution rate up to the second PC is more than 0.98, and so almost all the information is displayed in the plot. Six clusters corresponding to each of the targets are observed in the figure. This implies that each target has its own representation in the subspace regardless of its surrounding context, namely, an operation taking the target as its argument. This can also be stated as follows: a subspace holding information of a specific role, a target in this case, emerges. Furthermore, the clusters are arranged in a circle comparable to the continuum of color by hue. This arrangement suggests that the agent understands the similarity of color of the target. This is indirectly proven by the tendency to choose a target of a closely related but incorrect color. The more similar a dummy object is to a target with respect to color, the more easily the agent mistakes the dummy for the target. Therefore, the generalization of color is realized. When there is no specified target in the environment, the agent chooses an object of a similar color as a substitute.

In Fig. 5(b), the projection plane to see the differences among the operations is chosen by averaging the target information instead of the operation information. The accumulated contribution rate up to the second PC is more than 0.86. Here,

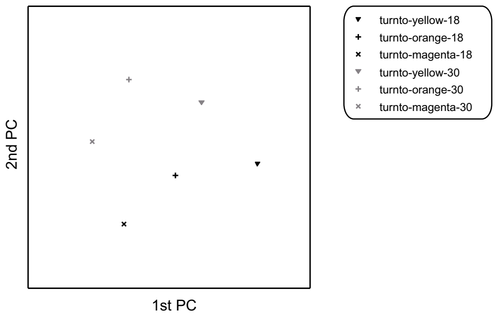

Fig. 4. A part of the concept space self-organized in experiment 2 is presented. PB vectors encoding actions constitute a geometric regularity as the underlying structure of the compositional system. PC means principal component.

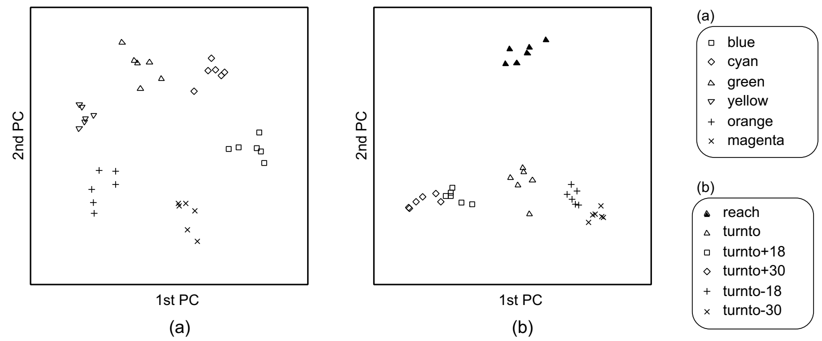

Fig. 5. The concept space self-organized in experiment 4. The space is projected to the planes in which the differences of 36 actions with respect to targets (a) and operations (b) are maximized.

both continuous and discrete sub-structures exist at the same time. In the first PC (x-axis of the figure), the continuum of `turnto` operations by offset emerges. Apart from that, the cluster of the `reach` operation is positioned. This implies that the second PC (y-axis of the figure) carries the distinction between `reach` and `turnto`. In addition, another continuum of the `turnto` operations by the absolute value of the offset is found in the third PC, the contribution rate of which is approximately 0.12. Thus, the subspace of operations consists of three orthogonal components. In addition, the subspaces of both targets and operations are orthogonal to each other, since the cosine between any pair of vectors taken from both subspaces is less than 0.12 ($cos^{-1}0.12 \simeq 83°$).

We discover at last an underlying analog mechanism of the phenomenologic system of symbols inferred in Section III by considering the following correspondences:

(1) The analog correspondence of an elemental symbol is the center of gravity of a cluster of actions containing the element as a part (see Figs. 5(a) and (b)).
(2) The composition of symbols is realized by summing up their corresponding vectors.

V. Discussion

We begin this section by discussing the incompatibility between the symbolic representation system and the usage-based model. The symbolic system failed to capture some important aspects of the usage-based scenario, as pointed out in Section I, although the symbolic system is scalable and advantageous in describing each developmental stage. Thus, we can never say symbolic representation is completely inadequate to describe the underlying mechanism of the learning and developmental process in a usage-based manner. Instead, a possible underlying mechanism that complements the conventional mechanism is investigated based on our experimental results. In other words, we present a mechanism of a sub-symbolic level. Both mechanisms can co-exist, much like the laws of chemistry are partly deduced from the laws of physics.

The essential difficulty of the conventional symbolic representation is a lack of mechanisms to properly represent a construction, which is a congruous pair of form and meaning. The symbolic model requires a predefined set of roles, but the introduction of a predefined set of roles causes a significant problem, namely, the discrepancy between form and meaning [6]. To state this problem clearly, we replicated the above experimental results with a model that employs a fixed set of roles. If the roles of a target, an operation, and an offset angle were given in advance of the learning, only one global compositional system could emerge in experiments 2, 3, and 4. Therefore, a mechanism other than the composition rules is required in order to emulate multiple different generalizations, including the partial compositionality observed in experiments 2 and 3. In fact, conventional symbolic models such as those described in [7], [8] employ grammatical rules that describe possible combinations of elemental concepts separately from the predefined semantic composition rules, although their models do not focus on usage-based theories. Here, two different rules for composition exist within a system. The first rule brings parts together into a whole, and the second rule is meaningless and provides only formal constraints. Thus, the discrepancy is generated when a symbolic model with fixed roles is used to explain the developmental process. Even worse, the discrepancy is easily ignored since the grammatical rules are often regarded as “composition rules.” In the following, we illustrate a dynamical systems interpretation of a usage-based model and show that the geometric nature of the representation restores the characteristic nature of a construction, which the conventional symbolic representation fails to capture.

A. From Similarity to “Symbol”

One of the essential benefits of the geometric representation is the compatibility of two different types of relationships among whole concepts: the sub-symbolic relationships based on surface resemblance and the symbolic relationships based on shared elemental concepts. As illustrated by Gentner in [9], a continuum between similarity comparisons and analogical comparisons based on shared structural properties can make it much easier to realize the analogical transfer of knowledge because the experience of similarity comparisons can facilitate the comprehension of structural properties through the continuum. A similar bootstrapping process is important in our model in order to accomplish recombination, which is a type of knowledge transfer.

Our bottom-up approach proceeds in the opposite direction of the conventional explanation in terms of the relationship between a symbol and the reusability of a part. In the usual sense of cognitive science, a predefined symbol is indispensable to explain a mechanism underlying recombination because the symbol represents a reusable unit. In the proposed approach, however, the reusability of acquired knowledge from the teaching data brings about the emergence of a symbol. The acquisition of a symbol is explained as the change of the reuse capability through learning. The improvement of the similarity measures of situations is important, because reusability refers to the capability whereby previously acquired knowledge is transferred to a novel but *similar* situation. As the similarity becomes focusing on structural properties more through the learning process, the reuse capability becomes more symbolic. Finally, then, a functionally compositional system emerges.

B. Essential Interdependency among Symbols

The agent achieves the recombination of goal-directed actions through the abovementioned bootstrapping process. The learning model of the action works as if it has a symbol corresponding to a conceptual element. It is, however, impossible to extract a single element from the conceptual space of the model. This means that the elements are acquired in a mutually dependent form. The interdependence among the elements is substantial in the domain of meaning since the roles of elements in a whole concept cannot be defined independently. For example, in an action consisting of two elements taking the roles of a target and an operation, both are defined in a circular manner. A target is something to be operated, and an operation affects a target in a certain manner. It is unnatural that the agent knows a general concept of a target without knowing any concrete operations. This is a typical example of the discrepancy between form and meaning. From the viewpoint of the usage-based cognition, it is plausible that a concept of a target changes as the agent acquires new operations on the target. On the other hand, we usually consider each elemental concept as atomic, namely, each elemental concept carries its own independent meaning. As employed in the conventional symbolic representation, a slot structure appears to be adequate for explaining the combinatorial generalization. Each slot is defined by its corresponding role, and its content can be switched independently from the other slots.

This contradiction is sublated in the geometric representation. The atomic element is implemented in terms of the independent subspace representing a conceptual role. The superficial independence of elements, however, strongly relies on the structure-preserving map between a PB vector and an action, which self-organizes in the connectivities of the network. The map provides all of the fundamental mechanisms to maintain the functional compositionality of the system, such as the composition rule and the semantic roles of elements. Without them, the conceptual elements cannot constitute a whole concept, just as chess pieces without a chessboard cannot constitute a game. Furthermore, the geometric representation inherently incorporates the interdependence among roles. Consider, for example, the simplest case depicted in Fig. 1. The role of an element is represented as the congruency of sub-structures: two congruent triangles for operations and three congruent sides for targets. Both congruent sub-structures always accompany each other through the geometry. Thus, we can state that a symbolic system focuses only on the superficial aspect. The abovementioned discrepancy between the meaning and form is generated because of the lack of attention to background mechanisms.

VI. Conclusion

We conclude that the functionally compositional system based on the proposed geometric representation is compatible with usage-based models. In addition, the geometric representation supports conventional symbolic models at a sub-symbolic level. The continuity between two relationships based on surface similarity and on shared reusable units plays an essential part in the emergence of a symbol. The interdependency among the semantic roles of concepts is also well illustrated in terms of the geometric nature of the relational structure. Moreover, the geometric representation may provide a longitudinal perspective, in which the transition between two contiguous stages is explained in terms of the continuous transformation of the internal structure. An experiment to support this idea will be conducted in the future. In addition, in the experiments presented herein, only the recombination of the internal representation is discussed but the combinatorial generalization of the associations between referents and sentences through the geometric representation will be investigated by employing the technique proposed in [10].

References

[1] B. Skinner, *VERBAL BEHAVIOR*. B.F.Skinner Foundation, 1957.

[2] M. Tomasello, *Constructing a Language: A Usage-Based Theory of Language Acquisition*. Harvard University Press, 2003.

[3] J. Pollack, "The induction of dynamical recognizers," *Machine Learning*, vol. 7, pp. 227–252, 1991.

[4] J. Tani, M. Ito, and Y. Sugita, "Self-organization of distributedly represented multiple behavior schemata in a mirror system: reviews of robot experiments using RNNPB," *Neural Networks*, vol. 17, pp. 1273–1289, 2004.

[5] T. van Gelder, "Compositionality: A connectionist variation on a classical theme," *Cognitive Science*, vol. 14, pp. 335–364, 1990.

[6] A. Goldberg, *Constructions: A Construction Grammar Approach to Argument Structure*. University of Chicago Press, 1995.

[7] D. Roy, "Learning visually grounded words and syntax for a scene description task," *Computer Speech and Language*, vol. 16, pp. 353–385, 2002.

[8] N. Iwahashi, "Robots That Learn Language: Developmental Approach to Human-Machine Conversations," in *Symbol Grounding and Beyond*, P. Vogt, Y. Sugita, E. Tuci, and C. Nehaniv, Eds. Springer-Verlag, 2006, pp. 143–167.

[9] D. Gentner and A. Markman, "Structure mapping in analogy and similarity," *Americal Psychologist*, vol. 52, no. 1, pp. 45–56, 1997.

[10] Y. Sugita and J. Tani, "Learning semantic combinatoriality from the interaction between linguistic and behavioral processes," *Adaptive Behavior*, vol. 13, no. 1, pp. 33–52, 2005.