

Symbol Statistics for Concept Formation in AI Agents

Jason R. Chen

*Department of Engineering, College of Engineering and Computer Science
Australian National University, Canberra, Australia
jason.chen@anu.edu.au*

Abstract—High level conceptual thought seems to be at the basis of the impressive human cognitive ability. Classical top-down (Logic based) and bottom-up (Connectionist) approaches to the problem have had limited success to date. We identify a small body of work that represents a different approach to AI. We call this work the Bottom Up Symbolic (BUS) approach and present a new BUS method to concept construction. The main novelty of our work is that we apply statistical methods in the concept construction process. Our findings here suggest that such methods are necessary since a symbolic description of the true agent-environment interaction dynamics is often hidden among a background of non-representative descriptions, especially if data from unconstrained real-world experiments is considered. We consider such data (from a mobile robot randomly roaming an office environment) and show how our method can correctly grow a set of true concepts from the data.

Keywords-cognitive architecture, concept formation, category, entailment, symbol statistics, bottom up AI

I. INTRODUCTION

The traditional approaches to AI are the top-down approach of Logic and Machine Learning, and the bottom up approach of Connectionism (Neural Networks). A key aim of both approaches is for agents with the ability to create and manipulate a high-level symbolic representation of their surrounding world. This mirrors the way that humans use high-level concepts to act intelligently in their environment.

Concepts are abstractions of past experience that allow the prediction of future outcomes. A concept like “tomato” allows a human to predict outcomes (taste, texture, etc) of a tomato not experienced before. A concept is (a) a collection of instances, called a category, and (b) its entailment. The instances within a category are all similar in some way, like our sensory experience of different tomatoes or the same tomato at different times in the above example. The entailment part of a concept is the consequence of category membership, and is what gives the concept its predictive ability when new instances of the category are encountered.

The traditional top-down and bottom-up approaches have had limited success to date. Indeed, it has even been conjectured that the top-down approach cannot form a solution to AI because its symbols are not grounded in sensorimotor experience [1]. In this paper, we identify a small body of work which represents an alternative approach [2], [3], [4], [5], [6]. We call it the Bottom-Up *Symbolic* (BUS) approach. This is to distinguish it from the Bottom-Up *Non-symbolic*

approach of Connectionism, given the non-symbolic nature of neurons. Essentially, the idea is to grow symbols that represent high level concepts in a bottom up fashion from an agent’s raw sensory and actuator data, without the restriction that the underlying substrate be Connectionist.

Of the BUS work, the 2 most similar to ours are [5], [4]. Work in [5] use different terminology in that they form “views” and “actions” as instances of similar sensorimotor experience, however the approach was tested in simulation only. Work in [4] constructs concepts as categories and entailments for a real robot that was programmed to precisely collide with a small set of distinct objects. The data set in which they searched for concepts representing these objects then contained only the concepts they were looking for. A concept formation framework suitable for a random roaming exploration of the environment (as we present below) we believe, is a more realistic scenario for BUS concept formation. Let us now expand on this point.

The categories of concepts should have a predictive inference on entailments. Finding concepts in sensorimotor data therefore means finding determinism in the relation between categories and entailments. Usually such determinism is hidden. The environment may be nondeterministic or dynamic [7]. Even for simple deterministic and static environments, the agent’s view of the environment may seem random because it is inaccessible [7] (inaccessible means the agent’s sensor set does not give it access to the complete state of the environment). Although the agent-environment interaction is deterministic, it then does not appear so to the agent. Further, real sensors and actuators suffer from noise. These issues mean our search for concepts for real-world agents and environments is a search for determinism in a seemingly random relation between the categories experienced by the agent and their entailments. In this paper we propose a framework for finding determinism in such a setting based on a method called Symbol Statistics [8].

II. FINDING CANDIDATE CONCEPTS

Our framework for growing concepts is envisaged for a wide range of applications, from a simple valve controller, to a complex humanoid robot. Such agents will have sensors, which are read by the agent, and which reflect the state of the environment. They will also have actuators, which are written to by the agent, and which operate on the environment. Assume that each sensor (actuator) returns

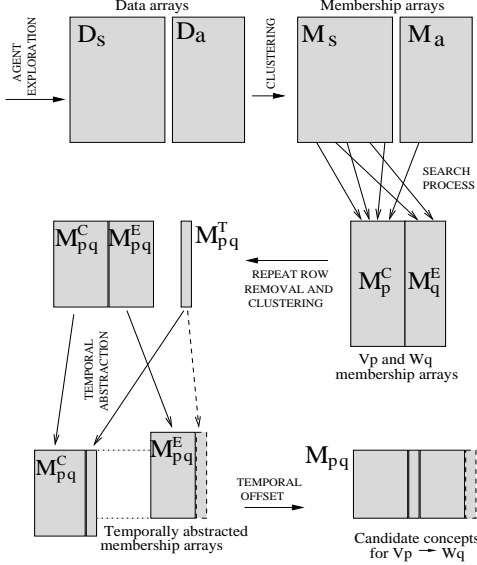


Figure 1. The Candidate Concept construction process

(receives) a scalar value at each point in time ¹. The idea is to deploy the agent in its environment with a basic (maybe random) set of behaviors such that exploration results. The agent will experience the consequences of particular sensory inputs and actuator outputs. We call these experiences *candidate* concepts. However, not all candidate concepts can be taken as true concepts, as we shall see. Prior to selecting true concepts in Section III, we detail (in this section) what candidate concepts are and how they are formed.

A. Where to Search for Candidate Concepts

Let $s = s_1, \dots, s_i, \dots, s_{ns}$ denote the agent's (scalar) sensor set and $a = a_1, \dots, a_j, \dots, a_{na}$ the agent's actuator set. Let S_i denote the set (range) of outputs for s_i and A_j the set (range) of inputs for a_j . Call S Sensor Space, $S = S_1 \times \dots \times S_i \times \dots \times S_{ns}$ and call A Actuator Space, $A = A_1 \times \dots \times A_j \times \dots \times A_{na}$. The space $S \times A$ then represents the full possibility of an agent's sensorimotor experience. Our definition of the category part of a concept as "instances of sensorimotor experience" means our search for categories will be in $S \times A$, or more generally, in a subspace of $S \times A$. If $\wp(s \cup a) = (v_p | p=1 \dots 2^{ns+na} - 1)$ denotes the power set of s union a , then our search must include one subspace V_p for each element v_p in the power set, where, if $v_p = (s_a, s_b, \dots, a_\alpha, a_\beta, \dots)$ then $V_p = S_a \times S_b \times \dots \times A_\alpha \times A_\beta \times \dots$. For example, a mobile robot with $s = (C)$, (C for Compass) and $a = (Lw, Rw)$ (Left and Right wheels), concept categories can potentially live in any of the subspaces V_1 to V_7 corresponding to $v_1 = (C)$, $v_2 = (Lw)$, $v_3 = (Rw)$, $v_4 = (C, Lw)$, $v_5 = (C, Rw)$, $v_6 = (Lw, Rw)$ and $v_7 = (C, Lw, Rw)$. Previous work [4] form categories from sensory data only, rather than sensorimotor data as we propose here.

¹Apparatus that deal in vector quantities at each time step, like laser range finders, can be viewed as sets of scalar returning apparatus

Our search for entailments differs to that for categories. The entailment part of a concept must live only in S or some subspace of S . The entailment infers predictive ability to the agent for some category of sensorimotor experience. The agent is generally interested in predicting the category's effect on the state of the environment, and its view of the state of the environment is encompassed by its sensory regime alone. Let $\wp(s) = (w_q | q=1 \dots 2^{ns} - 1)$. Then we must search for entailments in one subspace W_q for each element w_q of the power set.

The final step in this section is to bring together the category and entailment searches just specified. We will identify true concepts later as candidate concepts with a deterministic connection between the category and entailment. For each category space V_p , a deterministic entailment is possible in any of the entailment spaces W_q . Then the search for concepts means constructing candidate concepts across all combinations of $V_p \rightarrow W_q$.²

B. Clustering

We have addressed the issue of where to search for candidate concepts, but how should we construct them? Specifically, how should we form categories in V_p and entailments in W_q ? Recall that categories are groupings of sensorimotor experience that have similar characteristics. For example, [9] identifies different instances of the category "dog", such as Collies, Chihuahuas, Spaniels, etc. Each instance has similarities in their characteristics; they have fur, they pant, have four legs, etc, although they are not identical (their fur might be a different colour or length). This suggests that points close in V_p should form categories, and points close in W_q should form entailments. Such points represent instances of sensorimotor experience with similar characteristics, but which are not necessarily identical.

Clustering is a method used to identify groupings of closely spaced points in a metric space. For our application here, we could adopt the Euclidean metric in each V_p (W_q) and then cluster to find categories (entailments). However we propose an alternate approach of clustering first along the lines of sensor and actuator type, and then forming clusters in a higher dimensional V_p (W_q) by combining membership labels. We now explain this process in detail (see Figure 1).

Let D_s be our data set array ($n \times ns$) of the agent's sensor readings, where n is the number of data points recorded, and ns the number of sensors. Similarly, let D_a ($n \times na$) be our actuator data set. Columns in D_s represent the values returned by individual sensors, but there can be a number of columns in D_s that represent sensors of the same type, e.g. multiple bump sensors on a mobile robot. For each sensor type, cluster points in the relevant columns of D_s together. Repeat over all sensor types to return M_s ($n \times ns$),

²This search is exponential in ns and na . We formulate it in this way for now, however we will need to limit it, eg. by pruning unpromising sensor/actuator combinations, in future work

a membership matrix that holds the cluster membership label for each data point in D_s . Similarly, produce M_a from D_a on the actuator side. Clustering individual sensor (actuator) types together (rather than individual sensors (actuators) together) is appropriate since differences in the “spectra” of their data are due to the dynamics in the agent/environment interaction rather than physical differences in the hardware. We use fuzzy C-means clustering to produce M_s and M_a , and will explain why below. Once we have M_s and M_a , we form the membership matrix relevant to V_p . Call this matrix M_p^C , where the C superscript denotes a membership array for the category part of our candidate concepts. If $V_p = S_a \times S_b \times \dots \times A_\alpha \times A_\beta \times \dots$, then form M_p^C as the horizontal concatenation of columns a, b, \dots in M_s and α, β, \dots in M_a . Similarly, form M_q^E from columns of M_s for entailments. Rows in M_p^C (M_q^E) then have strings designating membership of data points in V_p (W_q).

While our approach to clustering requires a symbol string (rather than just a symbol) to represent each category (entailment) in V_p (W_q), it has the following advantages. It requires one up-front clustering computation (i.e. prior to the search specified in Section II-A) rather than a separate clustering computation for each V_p (W_q), and so is computationally attractive. It provides an intuitive labelling of what a category/entailment represents, e.g. wheel rotation data clustered into 3 clusters representing back, stop and forward rotation. The approach negates the need for a unit normalisation scheme across different dimensions of V_p (W_q) when furnishing the clustering space with the Euclidean metric. Finally, it is much easier to choose the number-of-clusters value for n separate single dimensional clusterings than for one clustering in an n -dimensional space.

Fuzzy C-means results in a degree of membership to a cluster for a data point, rather than the binary “is a member” or “is not a member” of hard clustering methods like K-means. Our use of fuzzy clustering has a pragmatic purpose, but has foundations in psychological models of human concept formation. The pragmatic purpose is to add hysteresis to category membership at boundaries of categories, since oscillation between categories can occur simply due to measurement noise in the data. For example, wheel velocity data of a mobile robot clustered into 3 clusters, 1 (fwd), 2 (no rotation) and 3 (back) may lead to a 1 to 2 transition of $\dots, 1, 1, 2, 1, 2, 1, 2, 2 \dots$. We use fuzzy clusters to identify the transition cleanly. *Boundary points* are identified by fuzzy C-means as those with a membership degree to both the pre- and post-boundary categories above a threshold. The choice of membership is then made by setting it equal to that of the preceding point in the trajectory. Work by Rosch [9] showed that, for humans, experiences can belong to multiple categories; essentially lying on a category boundary. For example, humans can categorise a tomato as a vegetable or a fruit. Their choice depends on the context; in the same way that a particular wheel velocity

might be categorised as forward rotation or little/no rotation, depending on the context of the recent past.

C. Temporal Abstraction and Offset

Clustering as just described is an abstraction step that transforms the specifics of a data point into a *type* of sensorimotor experience. A complementary abstraction step to this is temporal abstraction. That is, in addition to the type of sensorimotor experience, the dataset contains information regarding the duration of experience. Duration information in the dataset manifests itself as repeat rows in $M_p^C \oplus M_q^E$ (\oplus here represents horizontal concatenation of arrays - see Figure 1). Hence, our temporal abstraction step consisted of parsing $M_p^C \oplus M_q^E$ for repeat rows to create $M_{pq}^C \oplus M_{pq}^E$. For each set of repeat rows in $M_p^C \oplus M_q^E$, we added a row to $M_{pq}^C \oplus M_{pq}^E$ as the membership label string of this set. In addition we recorded the number of repeat rows in the set to create a count vector whose length was equal to the number of rows in $M_p^C \oplus M_q^E$. We then clustered the values in the count vector using K-means into (for this paper) 2 clusters: short and long duration, to produce a “temporal” memberships vector M_{pq}^T . The vector M_{pq}^T was then appended as the final column to M_{pq}^C (i.e. $M_{pq}^C \oplus M_{pq}^T$), to produce a set of membership labels for the data in V_p which reflected the temporal duration of the original strings in M_p^C . Temporal abstraction is also possible on the entailment side, and would capture the entailment duration aspect of the agent-environment interaction dynamics. To keep histograms in Section IV to a reasonable size, we did not implement it here (i.e. dashed arrow in Figure 1). Temporal abstraction in [4] consists of recording data during the agent’s exploration phase for only 5 seconds around distinctive sensor value changes (eg. bump sensor going from 0 to 1). While this reduces the amount of data to process, we question whether it really is temporal abstraction. We would prefer to retain this term for temporally abstracted categories/entailments formed on the basis of their temporal properties (like duration here) in the data.

The predictive aspect of a concept’s category on its entailment implies a *temporal offset* between categories and entailments, i.e. categories experienced at one time should be matched up with the entailment that follows. Hence, we form M_{pq} as the horizontal concatenation of the first $n-1$ rows of M_{pq}^C and last $n-1$ rows of M_{pq}^E (see Figure 1). M_{pq}^E then holds the candidate concepts for the $V_p \rightarrow W_q$ category/entailment combination in our search.

III. SELECTING TRUE CONCEPTS

The candidate concepts obtained in the previous section will not all be true concepts. First (Problem 1), the sensor/actuator combinations for V_p may not be coupled through the environment with the sensor combination for W_q , eg. $Lw \rightarrow Temp$: mobile robot left wheel actuator has no effect on room temperature. In essence, our “view” of the



Figure 2. Experimental Setup: Pioneer DX3 Robot roaming in a partitioned-off area of our lab

LwxRw → G	CxG → C	LaserxLwxRw → Bump
back, back (short) → noRot	north, aClock (short) → north	close, back, fwd (short) → bump
back, back (long) → noRot	north, aClock (short) → west	close, stop, back (short) → noBump
back, stop (short) → aClock	north, noRot (short) → north	close, stop, stop (short) → bump
back, stop (long) → aClock	north, noRot (long) → north	close, stop, fwd (short) → bump
back, fwd (short) → aClock	north, clock (short) → north	close, fwd, stop (short) → bump
back, fwd (long) → aClock	north, clock (short) → east	close, fwd, fwd (short) → bump
stop, back (short) → clock	east, aClock (short) → north	close, fwd, fwd (long) → bump
stop, stop (short) → noRot	east, aClock (short) → east	med, back, back (short) → noBump
stop, stop (long) → noRot	east, noRot (short) → east	med, back, stop (short) → noBump
stop, fwd (short) → aClock	east, noRot (long) → east	med, back, fwd (short) → noBump
fwd, back (short) → clock	east, clock (short) → east	med, stop, back (short) → noBump
fwd, back (long) → clock	east, clock (short) → south	med, stop, stop (short) → noBump
fwd, stop (short) → clock	east, clock (long) → south	med, stop, fwd (short) → noBump
fwd, stop (long) → clock	south, aClock (short) → east	med, fwd, back (short) → noBump
fwd, fwd (short) → noRot	south, aClock (short) → south	med, fwd, stop (short) → noBump
fwd, fwd (long) → aClock	south, aClock (long) → east	far, back, stop (short) → noBump
fwd, fwd (long) → clock	south, noRot (short) → south	far, back, fwd (short) → noBump
	south, noRot (long) → south	far, stop, back (short) → noBump
	south, clock (short) → south	far, stop, stop (short) → noBump
	south, clock (short) → west	
	south, clock (long) → west	
	west, aClock (short) → south	
	west, aClock (short) → west	
	west, noRot (short) → west	
	west, noRot (long) → west	
	west, clock (short) → north	
	west, clock (short) → west	
	west, clock (long) → north	

Table 1
TRUE CONCEPTS IDENTIFIED BY OUR FRAMEWORK

agent/environment interaction needs to be correctly formed. Second (Problem 2), noise and real world effects in the data mean the agent can experience candidate concepts not reflecting the true agent-environment interaction dynamics, even for the right “view” of the interaction.

We will focus in this paper on Problem 2. Our solution is based on a method called Symbol Statistics from the Dynamical Systems literature [8]. The idea is to partition a dynamical system’s state space into a set of labelled (with a symbol) partitions, thus enabling the system’s evolution to be described as a symbol sequence. A symbol tree containing the probability of experiencing different length symbol strings (words) is then constructed, where the descending levels in the tree correspond to words of length 2, 3, 4, etc. Work in [8] dealt with finding good models of dynamical systems by trying to match the symbol statistics of the experimental data with that of a model. A practical application of the approach on a real experimental system (an internal combustion engine) is presented in [10]. This work used a hypothesis testing technique called Shuffled Surrogates to provide confidence limits on the symbol statistics analysis, and we adopt this technique here to identify true concepts from our set of candidate concepts.

We took the same 4 basic steps as in [10] to construct the symbol statistics of our experimental data : partitioning, labelling, histogram construction, and shuffled surrogate calculation. The first 2 steps of the process (partitioning and labelling) were documented in Sections II-B and II-C, however we note some differences in our implementation of these steps compared to theirs. First, they construct a binary partition of their scalar data simply by identifying the median data value, whereas we used clustering as a practical method for generating a greater number of partitions. Second, while each of their partitions is labelled with a single symbol, ours have labels that are symbol strings (given our approach of pre-clustering sensors/actuators of the same type together and then forming partitions as strings of single-symbol membership designations). Third, their focus on dynamical systems means that they search for determinism in a map between domain and range spaces that are the same. Given our search here is for determinism between categories and entailments, the domain (V_p) and range (W_q) spaces were generally different. Fourth, a practical consequence of this last point is their interest in a range of levels in the symbol tree, whereas our interest here is only with level 2 (i.e. words of length 2). The end result from our partitioning and labelling steps is a sequence of 2 “letter” words (i.e. where the “letters” are symbol strings) as the rows in M_{pq} .

The histogram construction step is straight forward. Simply find the set of unique rows in M_{pq} and count the number of repeat occurrences for each row. The shuffled surrogate step follows. A shuffled surrogate data set is constructed by randomly shuffling the rows in $M_{pq}^C \oplus M_{pq}^T \oplus M_{pq}^E$ (i.e. the membership array obtained just prior to the Temporal Offset step in Figure 1). The symbol statistics of this shuffled data is obtained by performing the temporal offset step (from Section II-C) and the histogram construction step just detailed. The idea behind the shuffled surrogate analysis is that, if the agent’s experience in the environment contains significant sequence patterns (i.e. concepts), then the shuffling process will destroy these patterns and the symbol statistics of the shuffled data will differ to that of the original data. Any determinism in the original data will then be highlighted. A concrete measure of the amount of determinism present can be obtained by calculating confidence limits on the symbol statistics of the shuffled surrogates. If the shuffled surrogate process is repeated many (eg. here 1000) times the limits within which 95 percent of the statistics lie can be marked off. We select true concepts as candidate concepts whose frequency lies outside this 95 percent confidence band. True concepts whose frequency lie above the band represent *positive* true concepts in that their entailments describe what *will* occur given the category. Conversely, *negative* true concepts are those with a frequency lying below the 95 percent confidence band, and have entailments that describe what *will not* occur given the category. In what follows, we consider only positive true concepts.

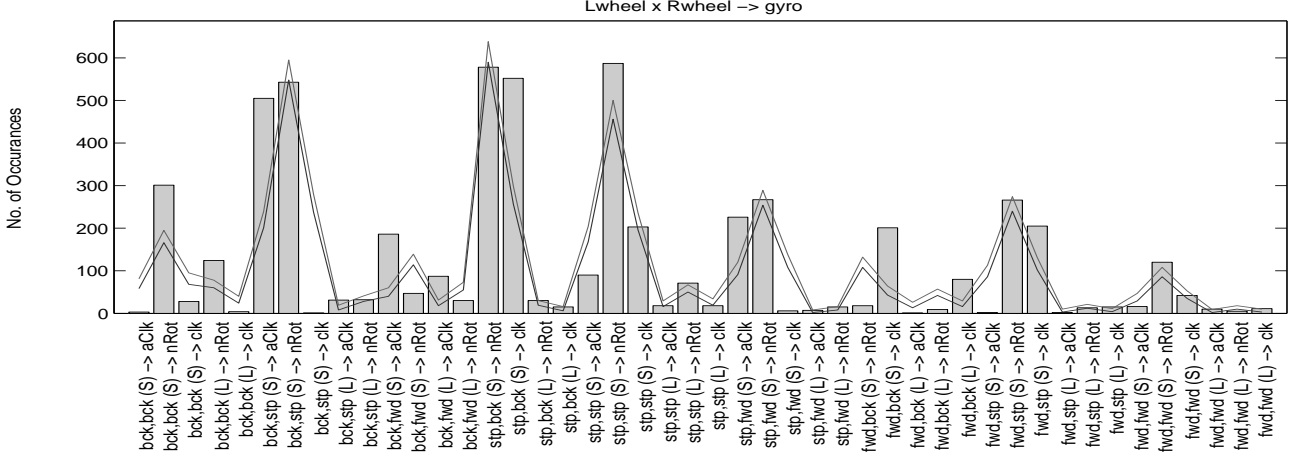


Figure 3. Word frequency histogram for $Lw \times Rw \rightarrow G$

IV. RESULTS/EXPERIMENTS

Experiments consisted of a Pioneer DX3 mobile robot allowed to roam randomly around a partitioned-off area of our lab (see Figure 2). The experiment lasted 3 hours, leading to 46077 points of recorded data for each robot sensor (compass (C), gyro (G), laser (L), bump (B)) and actuator (wheels: left(Lw), right(Rw)).

To test our method (given our focus on Problem 2) we manually chose three $V_p \rightarrow W_q$ combinations where true concepts should be found. The first was $Lw \times Rw \rightarrow G$, ie. could sensible concepts relating wheel commands and robot rotation be formed? Figure 3 shows a histogram of all candidate concepts found. Upper and lower line plots mark the 5 to 95 percent confidence intervals from shuffled surrogate analysis. Table I explicitly lists those identified as true concepts. Temporal abstraction here led to a reduced data set of 5622 points, and a “short” label with duration of up to about 3.5 seconds.

Broadly one can see that the true concepts selected make sense. For example, where left wheel rotation is more positive than right, a clockwise rotation results. The converse situation leads to an anticlockwise rotation. When both wheels have the same commanded velocity, a no-rotation entailment generally results. Curious are the Long Lw and Rw forward rotations, where either clockwise or anticlockwise rotation is predicted. This makes sense given the robot’s roaming behaviour consisted of independently adding/subtracting a small random amount to each wheel velocity at periodic time steps. Significant velocity differential was then possible, especially over long durations. It was at first puzzling why this did not also occur for long backward Lw and Rw rotations. The answer was because the roaming behaviour limited the maximum backward wheel velocity to something much less than the maximum forward velocity, so the differential velocity possible was then much smaller. Hence, the method does seem to be capturing the true dynamics of the agent-environment interaction.

An important outcome is that our method avoided candidate concepts in the histogram that are “unrepresentative” of the agent-environment interaction dynamics (17 true concepts were selected from 46 candidate concepts). For example, candidate concepts like *back-stop-(short) → Clockwise*, or *forward-stop-(short) → Aclockwise*, were experienced by the agent but do not correspond to an accurate symbolic representation of the underlying dynamics of the system. Rather, this type of candidate concept occurs due to the noise and imperfections that exist in any real world data, something corroborated by the fact that they were only ever experienced rarely and over short duration.

The second $V_p \rightarrow W_q$ combination we tested was $C \times G \rightarrow C$, ie. current heading and rotation should predict future heading. Table I shows 28 true concepts selected out of 46 candidate concepts. The temporal abstraction step here led to 3327 data points and a short duration label corresponding to less than about 6 seconds.

The true concepts selected in this case also make sense. For any heading, no gyro rotation (long or short) led to the same heading. When clockwise rotations occur, they either lead to the appropriate adjacent heading (e.g. $N \rightarrow E, E \rightarrow S$, etc.) or they lead to the same heading if the rotation is of a short duration (the latter makes sense since a short duration rotation may not move the heading outside its 90 degree range). The observations similarly make sense for anticlockwise rotations. It is again important to note that this set of sensible true concepts was selected from a much larger set of sometimes inappropriate candidate concepts. For example, candidate concepts such as *N-Clockwise-(short) → W*, or *S-Clockwise-(short) → E*, occur due to noise in the data and were rejected by our method (noise in compass data can be caused by stray magnetic fields, eg. from the robot’s own motors).

Collision is a key aspect of a roaming robot’s experience, so we used our method to look for concepts relating to collision. The robot has a laser range finder to measure

range to obstacles, and bump sensors located around its perimeter. To simplify the analysis we considered only frontal collision experiences, ie. $L \times Lw \times Rw \rightarrow B$ (so L is the forward direction laser reading and B the front bump). Table I lists 19 true concepts identified from 68 candidate concepts. Temporal abstraction here lead to 5164 points and a “short” label corresponding to less than 3 seconds.

The true concepts in Table I have extracted one important aspect of the collision dynamics; the laser must be returning a reading of “close” before a bump will occur. Of the true concepts with close as the initial laser reading, the key finding by the framework is that when both wheels are travelling forwards, a bump will occur. When the wheel rotations are mixed, but result in forward motion, then a bump is also correctly predicted. Only when the motion is backwards (e.g. *close-stop-back-(short)*) does a no bump outcome get predicted. These outcomes make complete sense and very well capture the dynamics present in the system. Obvious exclusions here are *close-back-back-(long or short) → nobump*. These candidate concepts were very close to lying outside the 95 percent confidence band, however they occurred much less than the *close-stop-back-(short)* category. This meant that the shuffled surrogate analysis was not quite able to determine if their occurrence was due to random chance or true determinism. Also requiring explanation is the inclusion of *close-stop-stop-(short) → bump*. Recall that *stop* will represent a range of motion including very slow wheel rotation. Since the robot could often be very close to an obstacle, even slow wheel rotation will then cause a bump to occur. Finally, the method proposes as deterministic a set of no-bump outcomes for initial laser readings of “far” and “medium”, and this is also consistent with the dynamics in the agent-environment interaction.

V. CONCLUSION

High level conceptual thought is at the basis of impressive human cognitive abilities. However, neither top down approaches like those based in Logic, nor the bottom up approach of Connectionism, have been successful to date. A small body of work exists that we have called the “Bottom Up Symbolic”(BUS) approach, and this work seems attractive given it potentially circumvents the problems with the 2 existing approaches: it can lead to grounded symbols that mean something to the agent, and the concept (symbol) formation process is not constrained to have neural networks as a substrate. In this paper we proposed a new BUS method for growing concepts from data.

We divided the problem to be solved into Problem 1 (finding sensor/actuator combinations leading to concise descriptions of the determinism in the data) and Problem 2 (given a sensor/actuator combination, finding concepts that reflect the true dynamics in the agent-environment interaction). We focussed here on Problem 2, and demonstrated that the method can result in accurate descriptions

of interaction dynamics for a number of different sensor-actuator combinations. A key finding was the importance of the statistical foundation of our approach, since the true symbolic description of agent-world interaction dynamics is often hidden for real world data. Other novelties compared to previous work were: a temporal abstraction process that led to temporal properties of the data being incorporated into concept categories and entailments, the use of fuzzy clustering in category and entailment formation to recognise that context seems to affect how humans classify their experience of the world, and concept category formation out of sensorimotor (rather than just sensory) experience.

Further work, in the shorter term, includes finding a solution to Problem 1. In the longer term, concepts produced by the work here could be described as quite low level, in that they capture interaction dynamics integrally related to sensorimotor apparatus. We would view them as similar to human concepts such as bitter (taste), smoky (smell), or crackle, e.g. from a fire (hearing). Human cognition includes higher level concepts, eg. “difficult”, “trustworthy” or “love”. In the longer term, our aim is to augment the framework presented here to construct more abstract, higher-level concepts that are further from the agent’s sensorimotor apparatus, i.e. possibly by combining the low level concepts discovered here.

REFERENCES

- [1] S.Harnad, “The symbol grounding problem,” *Physica D*, pp. 335–346, 1990.
- [2] P.Vogt, “Bootstrapping grounded symbols by minimal autonomous robots,” *Evolution of Communication*, vol. 4, pp. 89–118, 2000.
- [3] R.Sun and T.Peterson, “Some experiments with a hybrid model for learning sequential decision making,” *Information Science*, vol. 111, pp. 83–107, 1998.
- [4] M.Rosenstein and P.R.Cohen, “Continuous categories for a mobile robot,” in *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, 1999, pp. 634–640.
- [5] D.Pierce and B.J.Kuipers, “Map learning with uninterpreted sensors and effectors,” *Artificial Intelligence*, vol. 92, pp. 169–227, 1997.
- [6] A.Billiard and K.Dautenhahn, “Experiments in learning by imitation - grounding and use of communication in robotic agents,” *Adaptive Behaviour*, vol. 7, pp. 411–434, 1999.
- [7] S.Russell and P.Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.
- [8] X.Z.Tang, E.R.Tracy, A.D.Boozer, A.deBrau, and R.Brown, “Symbol sequence statistics in noisy chaotic signal reconstruction,” *Physical Review E*, vol. 51, pp. 3871–3889, 1995.
- [9] G.L.Murphy, *The Big Book of Concepts*. MIT Press, 2002.
- [10] C.E.A.Finney, J.B.Green, and C.S.Daw, “Symbolic time-series analysis of engine combustion measurements,” *SAE Paper No. 980624*, 1998.