Middle East Technical University
Department of Computer Engineering

Learning Context on a Humanoid Robot using Incremental Latent Dirichlet Allocation

Hande Çelikkanat, Güner Orhan, Nicolas Pugeault,
Frank Guerin, Erol Şahin, and Sinan Kalkan

METU-CENG-TR-2015-01

April 2015

Department of Computer Engineering
Middle East Technical University
İnönü Bulvarı, 06800, Ankara
TURKEY

This page contains a Turkish translation of the title and the abstract of the report. The report continues on the next page.

Artırımlı Örtülü Dirichlet Paylaştırımı ile İnsansı Robotlarda Bağlamın Öğrenilmesi

Hande Çelikkanat, Güner Orhan, Nicolas Pugeault,
Frank Guerin, Erol Şahin ve Sinan Kalkan

Bilgisayar Mühendisliği Bölümü
Orta Doğu Teknik Üniversitesi
İnönü Bulvarı, 06800, Ankara
TÜRKİYE

Öz

Bu çalışmada, bağlam olgusunu formalize ederek, bir robotun duyu-motor etkileşimleri üzerinde temellendirdiğimiz kavramlar cinsiden modellemekteyiz. Bu kavramları, insan beynindeki kavram ağı hipotezinden yola çıkarak, Markov Rasgele Alanı temelli bir ağ yapısı üzerinde temsil etmekteyiz. Bağlam olgusunun, böyle bir kavram ağının üzerinde, metin dökümanlarını modellemek için yaygın olarak kullanılan Örtülü Dirichlet Paylaştırımı yaklaşımındaki bir örtülü değişkene denk gelecek şekilde modellenebileceğini öne sürmekteyiz. Ayrıca, standart Örtülü Dirichlet Paylaştırımı yönteminin artırımlı bir varyasyonunu geliştirerek, (i) artırımlı bir şekilde öğrenecek, yani yeni etkileşimlerde bulundukça her şeyi baştan öğrenmesi gerekmeyecek, ve (ii) dünyada yeni bir bağlamın ortaya çıktığını farkettikçe, bu bağlamı da modele hızlı bir şekilde entegre edebilecek bir mimari önermekteyiz. iCub insansı robotu üzerinde yaptığımız deneylerle, bağlamın kavram ağı üzerinde modellenmiş olmasının da getirdiği avantajlarla, yaklaşımımızın beklenmedik koşullara uyumlu, çevrimiçi çalışabilen, ve gürbüz bir yöntem olduğunu göstermekteyiz. Uyumlu ve çevrimiçi özelliklerini, yeni etkişimlerle sonradan ortaya çıkan bağlamları kendiliğinden keşfedebilmesinden dolayı, gürbüz yapısını ise alakasız etkenlere karşı dayanıklı olmasına ve çok az sayıda etkileşimde bile başarılı bir şekilde öğrenebilmesinden dolayı öne sürmekteyiz. Son olarak, iCub insansı robot platformu üzerinde, öğrenilen bu bağlam bilgisinin, nesne tanıma ve planlama senaryolarında, nasıl etkili bir şekilde kullanılabileceğini göstermekteyiz.

Abstract

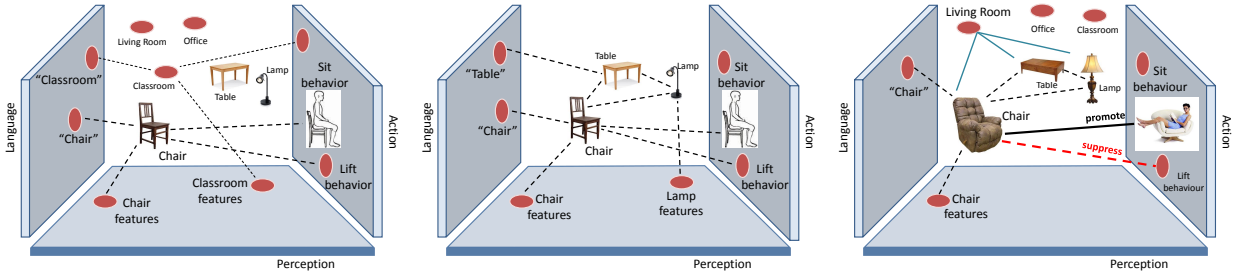
In this article, we formalize and model context in terms of a set of concepts grounded in a robot's sensorimotor interactions. The concepts are modeled as a web using Markov Random Field, inspired from the concept web hypothesis for representing concepts in humans. On this concept web, we treat context as a latent variable of Latent Dirichlet Allocation (LDA), which is a widely-used method in computational linguistics for modeling topics in texts. We extend the standard LDA method in order to (i) make it incremental so that it does not re-learn everything from scratch given new interactions (*i.e.*, it is online) and (ii) discover and add a new context into its framework when necessary. We demonstrate on the iCub platform that, partly owing to modeling context on top of the concept web, our approach is adaptive, online and robust: It is adaptive and online since it can learn and discover a new context from new interactions. It is robust since it is not affected by irrelevant stimuli and it can learn context after a few interactions only. Moreover, we show how iCub can utilize context learned in such a model for two important tasks: object recognition and planning.

1 Introduction

We tackle the problem of using contextual information to improve the performance of a cognitive robot, specifically in perception and planning. We define context as the totality of the information characterizing the situation of a cognitive system; *e.g.*, it can include objects, persons, places, and temporally extended information related to ongoing tasks, but also information not directly related to these tasks [16]. Our goal is to build a cognitive system which can learn the statistical associations between such items from experiencing many examples, and then use this information to help to identify objects at run-time, and to prune plans as appropriate to a situation.

There is ample evidence that natural cognitive systems modulate their response to stimuli depending on a wide range of other, seemingly irrelevant stimuli (context). A nice example of this in developmental psychology is where three month-old infants have been shown to be able to learn to move a crib mobile by kicking, but that they associate this behavior with the particular border surrounding the crib when the action is learned (a distinctive cloth draped over the crib rails in this experiment). If the border is the same as the original and infants are tested 3 or 5 days later, then the infants remember and repeat the behavior; if the border is changed the infants do not repeat the behavior (even though the mobile is the same) [12]. Moreover, Yeh and Barsalou [105] demonstrated in a series of experiments that human subjects perform better at a variety of cognitive tasks when taking context into account. This is because context can promote relevant information and behaviors, while suppressing irrelevant ones, based on statistical likelihood of various objects and behaviors in a certain setting. For example, Yeh and Barsalou [105] suggest that a concept such as a chair does not exist in isolation, but is associated in memory with other concepts that also occurred in the concrete situations where the concept was previously encountered by the system; e.g., the chair's location, office or living room, but also the actions performed with the chair such as reclining. These connections between concepts in memory allow then the system, when detecting concept, to draw inferences about connected concepts; this is illustrated in Figure 1. The activation of a 'chair' concept promotes related objects such as 'table' and 'lamp' and draws inferences on their plausible position. Furthermore, a 'living room' concept will promote chair properties such as 'large' and 'soft', rather than 'small' and 'hard' (contrary to, eg, a 'classroom' concept). Similarly, actions usually associated with the active concepts, such as 'sitting' in our example, are promoted whereas unlikely actions ('lifting') are suppressed. In sum, what forms context depends on the concept of interest, and consists of all other concepts present at the same time. Through experience, a cognitive system forms an interconnected network of related concepts and situations that allows efficient filtering of context and inference.

In this article, motivated from the concept-based nature of human cognition [80, 79, 26, 25, 34, 50, 56, 57, 17, 67, 46, 86, 76, 62, 68, 83, 89], we formulate context to be the set of active concepts in the scene rather than relating it to raw sensorimotor data. For this, we employ a widely-used topic model in computational linguistics, called Latent Dirichlet Allocation (LDA), and apply it to the active concepts in the scene. For modeling the concepts, we use a concept-web model that we developed using Markov Random Fields in our previous work [15]. We demonstrate how context can be learned and used by such a model for several tasks by a humanoid robot.



(a) Existing cognitive systems

(b) Our previous concept web model (not including context)

(c) Our proposal that links context with active concepts in the web

Figure 1: (a) Existing cognitive systems have concepts which have links to perceptual features and motor actions which were programmed by a designer or trained in context-free environments. (b) The concept web model that we developed in our previous study [15]: A densely connected concept web connecting perception, action and language; however, there is no notion of context in this model. (c) We propose a system that learns in context the links between concepts and sensorimotor primitives, based on the statistics of its interactions in real-life environments. For clarity, only a few links and concepts are shown. [Sub-figures (a) and (c) adapted from [16]]

1.1 Context in Cognitive Science and Robotics

It is a matter of consensus across fields that context processing is an essential part of embodied cognition (*e.g.*, psychology [105], language [8, 49, 21], AI [88], robotics [11, 63] and computer vision [97]. Schank and Abelson [88] argued that reasoning about situations in daily life relies on "scripts" that inform reasoners about the prototypical features of these situations. A restaurant, for example, tends to come with a menu, dishes, a waiter, a chef, and so on. This work has gone on to influence today's formal ontologies. Probably the earliest research on context focused on linguistic phenomena, studying how the understanding of an expression (*e.g.*, a personal pronoun like "it") is affected by the rest of the sentence or text [49]. Later research applied these ideas to other aspects of communication, including speech (*e.g.*, pitch accent) and body language (*e.g.*, [100]). Even more drastically, the notion of a context has been extended to all symbolic systems (*e.g.*, [2]). Perhaps most notably, McCarthy [63] proposed the rectification of context in classical (logical) AI, arguing that Artificial Intelligence needs to put the notion of a context centre stage. In McCarthy's view, intelligent machines "must construct or choose a limited context containing a suitable theory whose predicates and functions connect to the machine's inputs and outputs in an appropriate way" [64]. This work gave rise to a wave of theoretical work focusing on issues like the problem of "lifting" information about one knowledge base to another (*e.g.*, if a universally quantified proposition is known to be true in context X then what is the strongest proposition known to be true in a context Y that is related to X in some well- understood way).

Work in all these traditions continues to inspire Cognitive Science and AI. But times have changed: the rise of embodied cognition theories in the 90's, for instance, has offered a different perspective on context, based on a perceptual and action-based rather than symbolic approach [11]. This perceptual perspective is particularly relevant for robotics, where contexts typically need to be acquired from perception (*i.e.*, they cannot be programmed in advance). Barsalou, for example, has advocated the necessity for concepts to be situated [105, 7]; in other words, for an abstract concept to be related to concrete contexts. Coventry *et al.* [21] studied the difference between geometric and functional contexts in the use of spatial prepositions ("over" vs. "above") and of linguistic quantifiers ("few" vs. "many" vs. "several").

One striking example concerns work on affordances. Until some years ago, behavioral studies on affordances tended to highlight the fact that affordances are automatically activated, independently from the kind of task and context. This was shown through compatibility effects in which, for example, size resulted as a relevant dimension even if the task did not require subjects to judge objects on the basis of their size but, instead, of their category (*e.g.*, [99]). Recent evidence has questioned this view of affordances, showing that the activation of affordances is modulated by the physical and by the social

context. A variety of studies have shown that the embedding in a context given by a specific scene (*e.g.*, [47]) or by the presence of other objects influences affordances activation (*e.g.*, [106, 10, 71]).

Robotics has achieved significant success in terms of both theory and applications in the past five decades; however, research involving context has focused on the environmental aspect only, i.e., situation awareness, which involves perceiving and interpreting what is happening in the environment. Robotic studies have investigated situation awareness in urban search for rescue tasks [58], home security [36] and elderly people's living environments [98], object recognition in daily activities [3, 72], and trying to fulfill possibly incomplete natual language instructions of humans [66].

Of all these works, Anand *et al.* [3] and Misra *et al.* [66] stand out for attempting more explicit utilization of contextual information. Anand *et al.* [3] define and use a more restricted notion of context, limited to the spatial relationships between canonical placements of objects in the environment, used for object search and labeling. On the other hand, Misra *et al.* [66] treat context as multiple-choice values of the states of known objects in the environment (*i.e.*, microwave door is *closed* or *open*), used afterwards for completing missing information in natural-language commands of humans.

In computer vision, the notion of context has grown in prominence over the last decade, both explicitly and implicitly. Explicitly, the study of visual gist [74] showed that holistic encodings of the visual input could carry a large amount of information for intelligent systems allowing scene identification [74, 82], urban scene detection [78], and autonomous navigation [1]. Also, the importance of context in visual detection and recognition tasks has become prevalent in recent years: action recognition [61], object categorization [19], and detection [97]. Implicitly, the now popular data-driven, machine learning-based approach to vision led to algorithms that efficiently extract all predictive information from the visual data, effectively making heavy use of context to reach high performance (see [77] for a criticism).

A promising approach for developing an explicit model of context seems to be Latent Dirichlet Allocation (LDA), a hidden topic model developed for categorizing documents of large text corpora [9]. As a robust, unsupervised Bayesian method, it has been utilized recently in a variety of applications ranging from detecting "hot topics" in science [38], to fraud detection [104], activity profiling [33], and identifying functional regulatory networks of miRNAs and mRNAs [60]. Since the method provides the statistical tools for discovering hidden topics in unsupervised data, we propose that it can also be used for modeling context. In fact, ours is not the first attempt to use LDA formulation in robotics: It has been utilized successfully for object categorization from multi-modal sensory data [70, 69, 5], and for autonomous drive annotation [6]. However, our work is the first to use LDA for modeling context in robotics.

1.2 This Study

We see in existing works various piecemeal efforts to tackle particular facets of context in specific application domains. In contrast, and following the intuitions of [105], we argue that a principled approach is needed to learn, represent and process context in a developing cognitive system. If such can be achieved then the benefit will not be for just one task, but across all areas of cognition. However, such a computational or robotic implementation has not been attempted yet.

In this article, we study how we can equip a robot with the ability to detect and learn a context as well as use it for tasks such as object recognition and planning as proof of concept. The novelty and contributions of our approach can be summarized as follows:

- Formalization of context on a robot using Latent Dirichlet Allocation (LDA). To the best of our knowledge, this is the first time that context is tackled with systematically, or modeled *per se*, as a separate entity but also in direct relation with other conceptual entities, in a robotics scenario. In contrast to the attempts of Anand *et al.* [3] and Misra *et al.* [66] for using contextual information, which do not introduce a general model of context, resorting to defining it in terms of predefined geometric relations or object-part states, we formalize context and use this formalization to develop an adaptive system in which contextual information can be extracted, represented, and utilized explicitly.
- We provide an incremental extension of LDA so that (i) it does not re-learn everything from scratch given new sensorimotor interactions (*i.e.*, it is online) and (ii) it can discover and add a new context into its representations when necessary.

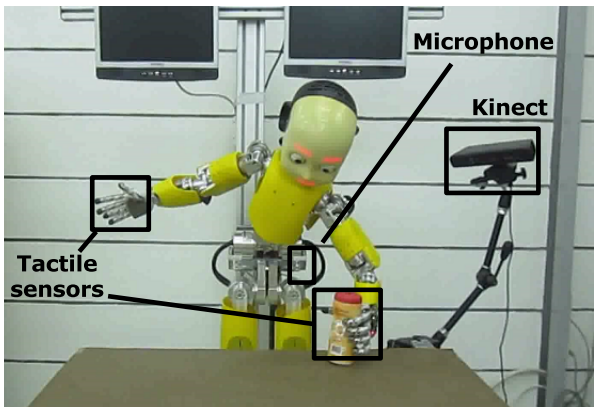


Figure 2: The setup used in the experiments. iCub senses the environment with its tactile sensors, a microphone and a Kinect.

- Finally, motivated by theories of a web of concepts in humans, and its possible advantage of dealing with the complexity of real-world scenes, we propose applying LDA on a concept web representation of the scene, instead of its raw features directly. We subsequently demonstrate how learning context from high-level abstract concepts is easier and achieves higher performance, compared to learning from raw features.

The current article extends an earlier version of our work [16], where preliminary results on integrating context were presented using the standard LDA with an ad hoc concept web. The current article differs in the following aspects: (i) The LDA is extended in order to make it online and incremental. (ii) The ad hoc concept web is replaced with a formally developed concept-web modeled using Markov Random Fields. (iii) A more extensive analysis of the system is presented.

The current article uses the concept web model that we developed before [15]. This previous work of ours introduced a concept web model and showed why it is important and useful. However, the current work goes beyond that and integrates context on top of that model to demonstrate how context can be learned and used by a humanoid robot.

2 Experimental Setup

We conduct our experiments using the iCub humanoid robot platform [65] (Figure 2). iCub has tactile sensors in each fingertip to detect the degree of grasping an object and gather the relevant information about the object's hardness. We utilize a Kinect device to get 3D information from the environment. iCub also has an external microphone to record the sound of objects.

2.1 Object Set

For the interactions, we have a set of 60 objects (call it $\mathcal{O}$) in total, which are arbitrarily divided into a training set (45 objects) and a testing set (15 objects). The training objects are labeled human supervision as belonging strictly to one of the 6 noun categories ($\{$*box, ball, cylinder, cup, plate, tool*$\}$ - see Figure 3) and one of the two adjectives in each 5 dichotomic adjective pairs ($\{$*hard* $\times$ *soft, noisy* $\times$ *silent, tall* $\times$ *short, thin* $\times$ *thick, round* $\times$ *edgy*$\}$ - see Figure 4).

The mapping between nouns and adjectives is not one-to-one; *e.g.*, a box can be soft or hard, silent or noisy etc. See Table 1 for the co-occurrence of the labels for the entire dataset.

2.2 Behaviors

We have a repertoire of 13 behaviors ($\{$*grasp, push left, push right, push forward, push backward, move left, move right, move forward, move backward, drop, throw, knock down, shake*$\}$). To ensure realism, some objects are (assumed to be) fragile and for this reason, certain behaviors are not applied on them:

(a) Balls (b) Boxes (c) Cylinders (d) Cups (e) Tools (f) Plates

Figure 3: The objects for each noun category.

(a) Hard (b) Noisy (c) Tall (d) Thin (e) Round

(f) Soft (g) Silent (h) Short (i) Thick (j) Edgy

Figure 4: The objects for each adjective category.

We prevent iCub from performing *drop*, *shake*, *throw*, *knock down* and *push* behaviors on plates and cups. We also refrain from applying *push* behaviors on balls, since they tend to roll down and disappear from the table when pushed. The applicable behaviors for objects with respect to their noun categories are shown in Table 2.

2.3 Features and Data Collection

iCub interacts with each object $o \in \mathcal{O}$ as follows:

1. An object o is put on the table.
2. iCub "looks" at the object (*i.e.*, takes a 3D snapshot using the Kinect sensor) and extracts the initial visual features $\mathbf{e}_v$.
3. While the *grasp* behavior is in progress, audio ($\mathbf{e}_a$), haptic ($\mathbf{e}_h$), and proprioceptive ($\mathbf{e}_p$) features are collected.
4. iCub takes a second 3D snapshot and extracts the final visual features $\mathbf{e}'_v$.
5. The object is placed to a different initial position (to allow possible variability) by a human supervisor, before proceeding with the next action.

Table 3 lists the features used by iCub in this study. The first 7 visual features are basic position information and three dimensional properties of the object, and the next 40 features are the zenith and azimuth normal vectors of each point on the object. In addition to the normal information, we use histogram of shape index values. Shape index [54] is essentially a representation of the local surface type, calculated from the maximum and minimum principal curvatures ($\mathcal{Q}_1, \mathcal{Q}_2$, respectively) of the point as follows: $\frac{\mathcal{Q}_1+\mathcal{Q}_2}{\mathcal{Q}_1-\mathcal{Q}_2}$.

The following 13 are auditory features ($\mathbf{e}_a$) used to determine whether an object produces sound when interacted with. We use MFCC (Mel-Frequency Cepstrum Coefficients) on the raw audio data, yields a set of 13-feature vectors. As features we use the differences between the maximum and minimum values of each vector.

Haptic and proprioceptive features ($\mathbf{e}_h$ and $\mathbf{e}_p$) are obtained from the index finger of iCub only. They are collected through the grasping action, and encode the difference between initial and final

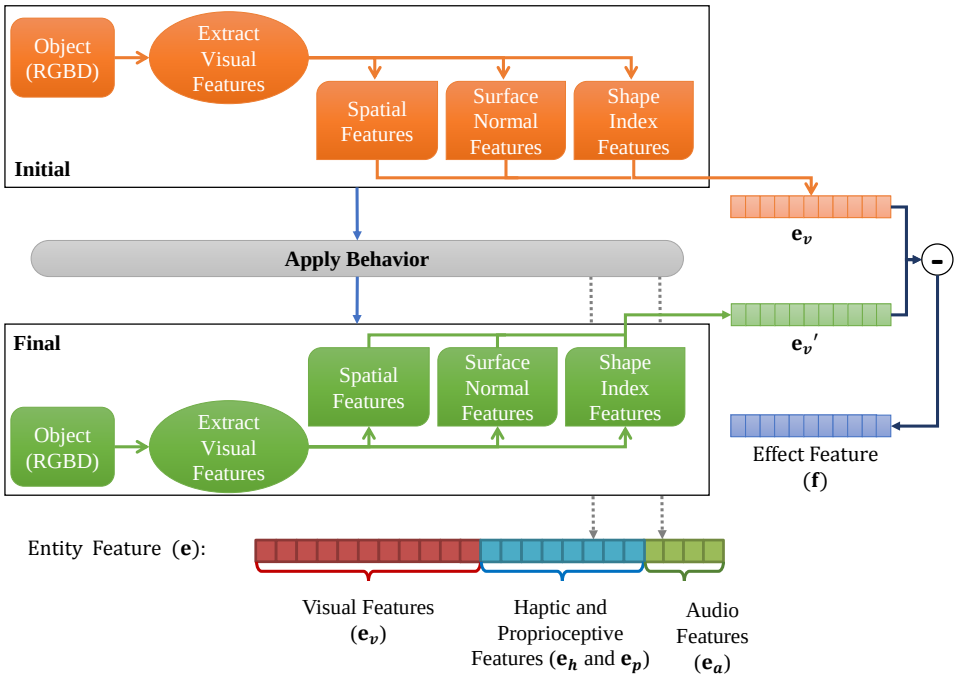


Figure 5: Extraction of entity features and effect visual features. $\mathbf{e}_v$ and $\mathbf{e}'_v$ are the visual features of an object before and after a behavior is applied. $\mathbf{f} = \mathbf{e}'_v - \mathbf{e}_v$ is the effect visual feature. $\mathbf{e}$ is the multi-modal feature incorporating visual, haptic, proprioceptive, and audio information.

Table 1: The co-occurrences of the noun and adjective labels for the entire dataset. The mapping between nouns and adjectives is not one-to-one. Numbers denote the number of objects (out of 60 objects in the dataset) sharing the noun and adjective categories.

	Hard	*Soft*	*Noisy*	*Silent*	*Tall*	*Short*	*Thin*	*Thick*	*Round*	*Edgy*
Box	2	14	2	14	0	16	0	16	0	16
Ball	3	7	7	3	0	10	1	9	10	0
Cylinder	14	0	5	9	10	4	9	5	14	0
Cup	11	0	1	10	0	11	0	11	11	0
Tool	5	0	5	0	5	0	0	5	5	0
Plate	4	0	0	4	4	0	0	4	4	0

Table 2: The set of behaviors applicable for each object. A:*Applicable*; NA: *Not-Applicable*

	Push (*Left, Right, Forward, Backward*)	*Move* (*Left, Right, Forward, Backward*)	*Drop*	*Grasp*	*Shake*	*Knock down*	*Throw*
Box	A	A	A	A	A	A	A
Ball	NA	A	A	A	A	A	A
Cylinder	A	A	A	A	A	A	A
Cup	NA	A	NA	A	NA	NA	NA
Tool	A	A	A	A	A	A	A
Plate	NA	A	NA	A	NA	NA	NA

sensor readings for haptic/proprioceptive data, the minimum and maximum readings, and also the mean, variance, and the standard deviation values.

The concatenation of these features $(\mathbf{e}_v, \mathbf{e}_a, \mathbf{e}_h, \mathbf{e}_p)$ is called an *entity feature vector* and is denoted by $\mathbf{e}$. Each object is described by an entity feature. For describing behaviors, we use *effect feature vectors*, denoted by $\mathbf{f}$, capturing the effect of a behavior on an object. They give the difference between the visual feature of the object $(\mathbf{e}'_v)$ after and before a behavior is applied, obtained by $\mathbf{f} = \mathbf{e}'_v - \mathbf{e}_v$. See Figure 5 for an illustration.

Table 3: The visual, audio, haptic and proprioceptive features extracted from the interactions of the robot

Feature Type	Feature	Position
Visual ($\mathbf{e}_v$)	Position:(x, y, z)	1-3
	Object dimensions:$(width, height, depth)$	4-6
	Normal zenith histogram bins	7-26
	Normal azimuth histogram bins	27-46
	Shape index histogram bins	47-66
Audio ($\mathbf{e}_a$)	13 bins of MFCC (max - min)	67-79
Haptic ($\mathbf{e}_h$)	Change for index finger	80
	Min values for index finger	81
	Max values for index finger	82
	Mean for index finger	83
	Variance for index finger	84
	Standard deviation for index finger	85
Proprioceptive ($\mathbf{e}_p$)	Change for index finger	86
	Min values for index finger	87
	Max values for index finger	88
	Mean for index finger	89
	Variance for index finger	90
	Standard deviation for index finger	91

Table 4: Used contexts and their prevalent concepts.

Kitchen			**Playroom**			**Workshop**		
cup	short	thin	ball	edgy	silent	tool	edgy	tall
plate	hard	thick	box	soft	thick	cylinder	hard	thin
round	silent		round	noisy		round	silent	thick

2.4 Contextual Setting

Our experimental setting is comprised of three contexts, *Kitchen*, *Playroom*, and *Workshop*. Some concepts in our framework occur in certain contexts, such as plates and cups existing in a Kitchen, balls and boxes occuring in a Playroom, as so on. Notice that this tendency is mostly a characteristic of noun concepts, which have more clear-cut divisions into contexts. On the contrary, some concepts are so general that they do not have such clear-cut divisions. This is a characteristic of most adjective concepts in our setting: Adjectives such a round or tall are so generic that they are not limited to certain contexts. Table 4 summarizes the prevalent concepts of the three contexts.

3 A Concept Web Using Markov Random Field

In our system, context is formalized over a set of concepts that are extracted from the scene, and represented in a densely-connected web structure, called the concept web [15]. Since this web is central to our model, before continuing with the exact formalization and use of context in the system, we briefly describe the extraction of relevant concepts from a scene, and the formation of the concept web. We describe a framework consisting of three kinds of concepts: Noun concepts $\mathbb{N}$ ={*box, ball, cylinder, cup, plate, tool*}, adjective concepts $\mathbb{A}$ ={*hard* $\times$ *soft, noisy* $\times$ *silent, tall* $\times$ *short, thin* $\times$ *thick, round* $\times$ *edgy*}, and verb concepts $\mathbb{V}$ = {*grasp, push left, push right, push forward, push backward, move left, move right, move forward, move backward, drop, throw, knock down, shake*}.[1] Before evaluating each scene in terms of its context, the robot views and possibly interacts with the objects, makes initial predictions about the concepts associated with the scene, and then builds a web of these concepts to make use of

[1]Note that "verb concepts" do not have to correspond to the behavior set in a 1-1 manner: A verb concept can be associated with multiple behaviors, for instance, provided that all of these behaviors produce the same effect [48], although this is not the case in this study.

their related semantics.

3.1 Reasoning with Individual Concepts

The initial task of the robot is to predict the individual concept(s) that are related to an object in its environment. This mapping of the world from raw features to a concept can be learned in a variety of manners, *e.g.*, using Support Vector Machines [20], k-Nearest Neighbors [22], Neural Networks, etc. In this work, we adopt a prototype-based approach [84, 32] following previous work [48, 75]; however, this choice is not central to the rest of the article; any method that provides a measure of similarity to a category from raw features is sufficient for this part. For a review of alternative representation schemes, the reader may for instance refer to [30, 55, 85].

In our framework, we describe the concepts in terms of their prototypes, and the similarity of a given feature vector $\mathbf{x}$ to a concept c is determined by the distance $D(c, \mathbf{x})$ between $\mathbf{x}$ and the prototype of c:

$$D(c, \mathbf{x}) = \frac{1}{|\mathcal{R}_c \setminus \mathcal{R}_c^\star|} \sqrt{\sum_{i \in \mathcal{R}_c \setminus \mathcal{R}_c^\star} (\mathbf{x}^i - \mu_c^i)^2}, \tag{1}$$

where $\mathcal{R}_c^\star$ is the set of indices that are not relevant for concept c; $\mathbf{x}^i$ is the i^{th} dimension of $\mathbf{x}$, and μ_c is the prototype of concept c. For the complete procedure of prototype extraction and concept assignment, see Appendix A and [48].

3.2 From Individual Concepts to a Densely Connected Web

In [15], motivated by the findings supporting a web of concepts hypothesis in humans (see, *e.g.*, [80, 79, 26]), and the potential computational benefits of the approach, we combined individual concepts into a web using Markov Random Field (MRF) [52]. For the sake of completeness, we describe the method here briefly.

A Markov Random Field is an undirected graph of random variables, over which inference is often carried out by a minimization of a predefined energy function. In the energy function, the consistency of the categories (*i.e.*, the nodes) with the input (called the data term in MRF) and the consistency between the categories (called the smoothness term) are specified. By minimizing this energy function, an MRF finds the most likely categories for an input, satisfying also the regularization constraints specified in the smoothness term.

In our representation of concept web as an MRF, the nodes correspond to concepts, and the co-occurring concepts are connected via edges. With $\mathbb{C} = \mathbb{N} \cup \mathbb{A} \cup \mathbb{V}$ being the set of all concepts, the concept web W is defined as a graph, $W = G(\mathbb{C}, \mathbb{E})$, with each concept $c \in \mathbb{C}$ being a node in W, and edge $\epsilon_{ij} \in \mathbb{E}$ if concepts c_i and c_j co-occurred in the training set. In other words, the edges between the nodes (concepts) are learned from the training data, presented as individual objects, and actions on them.

What happens when a new observation arrives is depicted with a schematic representation in Figure 6. The edges from the input to the nodes correspond to the data term (represented in terms of ψ_C,

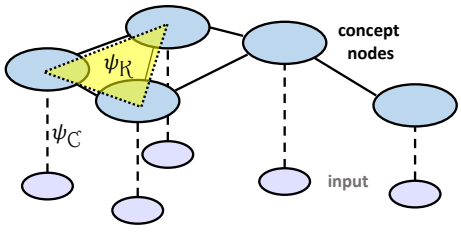


Figure 6: A schematic represensation of MRF modeling of the concept web. Initial predictions about the concepts are used to initialize concept node probability values. Conformance to initially predicted values are maintained by minimizing the sum of unary potential functions ψ_C. Meanwhile, clique potentials are initialized from the cooccurence information from the training data, and conformance to the cooccurence information is maintained through minimizing the sum of clique potentials ψ_K.

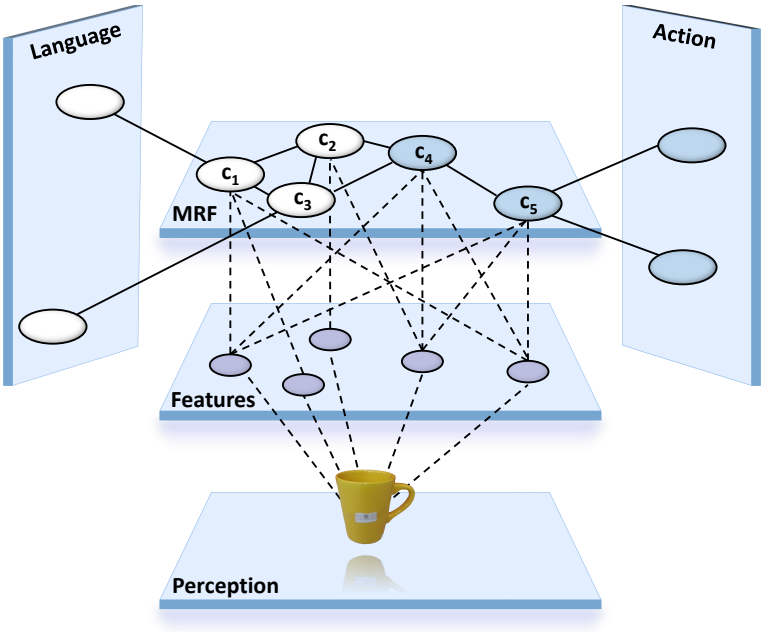


Figure 7: The schematic presentation of the whole system. Information can flow in from the perception space, through a feature extraction mid-level, or from the language and action spaces as well. At the end of MRF formation, the relevant concepts will be activated and connected to their counterparts in the three spaces. A number of nodes are randomly illustrated with white color to exemplify active concepts.

unary potentials), and the edges between the nodes model the smoothness term (represented in terms of ψ_K, clique potentials). The energy function $U(\omega)$, composed of these two terms, is minimized to find the most likely MRF configuration ω^*:

$$\begin{aligned} \omega^* &= \underset{\omega}{\operatorname{argmin}}\, U(\omega) \\ &= \underset{\omega}{\operatorname{argmin}} \left(\sum_{c \in \omega} \psi_C(c) + \sum_{\mathscr{K} \in \mathbb{K}} \psi_K(\mathscr{K}, \omega) \right) \end{aligned} \tag{2}$$

where the first term, *i.e.*, the data term, is a summation of the unary potentials for each active concept c in ω, and the second term is the smoothness term, as a summation of clique potential functions. The unary potential function denoted by ψ_C is defined as:

$$\psi_C(c) = D\left(c, \mathbf{x}\right), \tag{3}$$

with $\mathbf{x}$ being the instantaneous observation, $D(c, \mathbf{x})$ its distance to concept c (Equation 1). The potential function for cliques, denoted by ψ_K, is defined as:

$$\psi_K(\mathscr{K}, \omega) = \mathcal{V}(\mathbf{c}_{\mathscr{K}}), \tag{4}$$

where $\mathcal{V}(\mathbf{c}_{\mathscr{K}})$ is the potential of the clique $\mathscr{K}$ consisting of concepts $\mathbf{c}_{\mathscr{K}}$ active in ω, and $\mathbb{K}$ is the set of all cliques.

We minimize the energy function in Equation 2 using the Loopy Belief Propagation (LBP) algorithm [93, 43, 35] designed specifically for cyclic MRFs. A schematic depiction of the complete system is presented in Figure 7, showing the information flow from perception space through a feature-extraction mid-layer, as well as from language, and action spaces. The finalized concept web has all the relevant concepts in the active state (indicated with white color), and connected to their relevant counterparts in the three spaces. A sample concept web that is constructed by the iCub is shown in Figure 8.

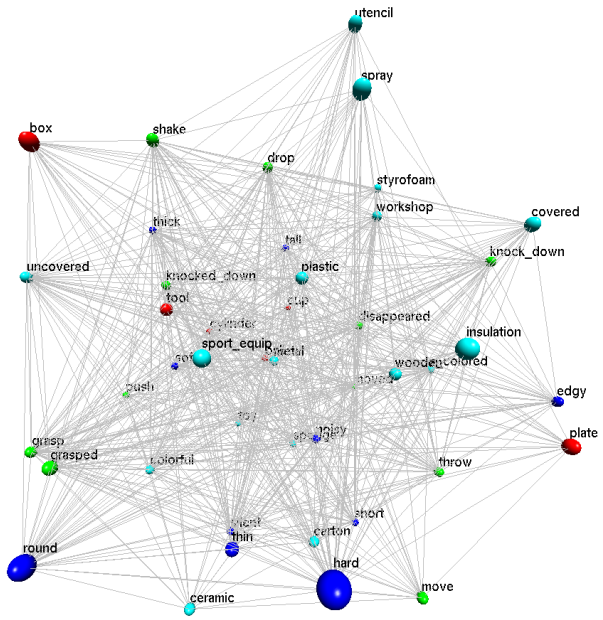


Figure 8: A sample concept web constructed by the iCub. Noun, adjective, and verb concepts are indicated with red, blue, and green respectively. Connections between concepts are shown with gray. Ubigraph visualization library is used for displaying the graph [102]. [From [15]. Best viewed in color.]

4 Formalization of Context

In this article, we claim that context is tightly related to concepts. When we see, *e.g.*, an environment with a sink, a dishwasher, and a table with cups and plates, we interpret the setting as a "kitchen". In this case, what triggers the interpretation of "kitchen"ness, the kitchen context, is the concepts of sink, dishwasher, etc [105]. A context can be triggered by object-related concepts (nouns, adjectives), as in this example, but also by verb concepts (*e.g.*, pouring), spatial concepts (*e.g.*, cups being on the table), temporal concepts (*e.g.*, morning, noon) or social concepts (*e.g.*, family, date). Let us denote all these types of concepts by $\mathbb{T}$ and define $\mathbb{T}$ as follows:

$$\mathbb{T} = \{t^{noun}, t^{adj}, t^{verb}, t^{adverb}, t^{spatial}, t^{temporal}, t^{social}\}. \tag{5}$$

Then, let us use $\mathbb{C}^t$ to denote the set of concepts of type t, with $t \in \mathbb{T}$. From this definition, it follows, for example, that the set of noun concepts, $\mathbb{N}$ (introduced in Section 3), is the same set as $\mathbb{C}^{t^{noun}}$. Moreover, let the set of all concepts be denoted with $\bigcup_{\forall t \in \mathbb{T}} \mathbb{C}^t$, and its power set with $\mathbb{P} = \mathbb{P}\left(\bigcup_{\forall t \in \mathbb{T}} \mathbb{C}^t\right)$.

The link between contexts and concepts might be of different types. For example, there are certain concepts related to a context specifically, such that their existence in a scene automatically invokes the related context. A dishwasher is a typical example, whose activation alone is enough to activate the kitchen context. In other cases, a *set* of concepts may need to be active together in order to invoke the context, such as water and boiling, which separately do not necessarily invoke the kitchen idea, but together do. This leads to the following definition:

Definition 1. *There exist concept sets $\mathbb{S} \in \mathbb{P}$ that sufficiently imply a context χ_k, that is,*

$$\exists \mathbb{S}\ (\texttt{Active}(\mathbb{S}) \implies \texttt{Active}(\chi_k)). \tag{6}$$

These sets are minimal *in that any proper subset of them does not necessarily trigger context χ_k:*

$$\forall \mathbb{S}_s \subset \mathbb{S}\ (\mathbb{S}_s \not\Longrightarrow \texttt{Active}(\chi_k)). \tag{7}$$

We call any such set an enforcing concept set of context χ_k, since it enforces the activation of context χ_k, and denote it with $\mathbb{S}^{+k}$. Since there are more than one such sets, let us use $\mathbb{P}^{+k}$ to denote the set of all such sets:

$$\mathbb{P}^{+k} = \{\mathbb{S} \mid (\texttt{Active}(\mathbb{S}) \implies \texttt{Active}(\chi_k))\}. \tag{8}$$

Not all concepts trigger a context. There exist concept sets that are in conflict with a specific context. A pool, for instance, is in conflict with the kitchen context. Such conflicting concept sets enforce the activation of an alternative context as defined below:

Definition 2. *There may exist concept sets $\mathbb{S} \in \mathbb{P}$ which are in conflict with context χ_k, and therefore enforce the activation of an alternative context $\chi_{\overline{k}}$:*

$$\texttt{Active}(\mathbb{S}) \implies \texttt{Active}(\chi_{\overline{k}}), \quad \chi_{\overline{k}} \neq \chi_k. \tag{9}$$

We call these conflicting concept sets of χ_k, and denote them with $\mathbb{S}^{-k}$. Since there are more than one such sets, let us use $\mathbb{P}^{-k}$ to denote the set of all such sets:

$$\mathbb{P}^{-k} = \left\{\mathbb{S} \mid \texttt{Active}(\mathbb{S}) \implies \texttt{Active}(\chi_{\overline{k}}), \chi_{\overline{k}} \neq \chi_k\right\}. \tag{10}$$

Real scenes might contain several contexts simultaneously. We may find ourselves in a studio flat with a combined kitchen-living room. Or in an outdoor bar next to a pool. In such cases, more than one context can be activated simultaneously in our minds, with all the implications due, such as the possibility of preparing a drink in the outdoor bar, together with the danger of falling into the pool. Therefore, contexts are not mutually-exclusive. This kind of multiple contextual activation is possible if *enforcing* concept sets of a context co-occur with its *conflicting* concept sets:

Property 1. If enforcing concept sets $\mathbb{S}^+_k$ of a context χ_k co-occur with its conflicting concept sets $\mathbb{S}^-_k$, both χ_k and and alternative context $\chi_{\overline{k}}$ are activated, χ_k due to $\mathbb{S}^+_k$, and $\chi_{\overline{k}}$ due to $\mathbb{S}^-_k$.

Definition 3. *If at least two different contexts are active in a scene ($\texttt{Active}(\chi_k) \wedge \texttt{Active}(\chi_{\overline{k}}) \wedge \overline{k} \neq k$), the scene is called a* mixed-context *scene.*

Not all concepts related to a context are enforcing in the sense given in Definition 1. For instance, a cup concept is *consistent*, *i.e.*, meaningful, in a kitchen context, but it alone cannot trigger the kitchen context. It can as well exist in a living room context, or in an office context. However, when surrounded with a sink and a dishwasher, a cup will also be thought as part of a kitchen context. This distinction yields the following definition:

Definition 4. *The remaining concept sets $\mathbb{S} \in P \setminus (\mathbb{S}^{+k} \cup \mathbb{S}^{-k})$ do* not *enforce context χ_k,*

$$\texttt{Active}(\mathbb{S}) \not\implies \texttt{Active}(\chi_k), \tag{11}$$

however, when considered together with enforcing sets $\mathbb{S}^+_k$, they are consistent *with the activation of context χ_k,*

$$\texttt{Active}(\mathbb{S}^+_k) \wedge \texttt{Active}(\mathbb{S}) \implies \texttt{Active}(\chi_k). \tag{12}$$

We call these consistent concept sets of χ_k, and denote them with $\mathbb{S}^{*k}$. Since there are more than one such sets, let us use $\mathbb{P}^{*k}$ to denote the set of all such sets:

$$\mathbb{P}^{*k} = \mathbb{P} \setminus (\mathbb{P}^{+k} \cup \mathbb{P}^{-k}). \tag{13}$$

From the definitions of the different types of concept sets that might be related to a context, we can now formally define a context as follows:

Definition 5. *A context χ_k, indexed by k, is a latent variable, which becomes activated if an enforcing concept set $\mathbb{S}^{+k} \in \mathbb{P}^{+k}$ is active.*

In summary, we deduce that a context has three different relations with concepts: (1) The set of enforcing sets of concepts, which necessarily invoke the activation of the concept, (2) The set of consistent sets of concepts, which do not necessarily invoke its activation, but are also meaningful in it and do not necessarily invoke the activation of an alternative context either, and (3) The set of conflicting sets of concepts, which are not meaningful in the context, and therefore necessarily invoke the activation of an alternative context.

Any attempt for modeling context must therefore be able to incorporate these properties of context. In this work, we present such a modeling of context, using Latent Dirichlet Allocation, which can explicitly handle all these properties.

5 Modeling Context Using Incremental Latent Dirichlet Allocation

In our framework, context is linked to the set of concepts that the robot perceives from its immediate environment. We use Latent Dirichlet Allocation (LDA) to detect the latent (unobserved) context(s) of the scenes. The observations of the robot are represented in terms of a concept web (Section 3), which is then used for analyzing context with the help of co-occurrence information. The detected context(s) are in turn fed back to the concept web in order to guide and correct its reasoning, similar to the feedback loops from higher-level cortices in humans. In this section, we provide the details of these steps.

5.1 Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) [9] is a method for modelling topics of documents in large text corpora. Assuming a document $d \in \mathbb{D}$ is a set of words $w_1, ..., w_N$ drawn from a fixed vocabulary ($w_i \in \mathbb{W}$ for vocabulary of size $|\mathbb{W}|$, where $|.|$ denotes set cardinality), LDA posits a finite mixture over a fixed set of topics $z_1, ..., z_k$ ($z_t \in \mathbb{Z}, |\mathbb{Z}| = K$ is the topic count). Then, a document can be described by the probability of relating to these topics, $P(z_t|d_i)$. Conversely, a topic is modelled by the likelihood for a document of this topic to contain each word in the vocabulary, $P(w_j|z_t)$. LDA proposes to infer these document and topic probability distributions from a set of documents $\mathbb{D}$.

Formally, LDA assumes a given set of documents, called a corpus, has originally been generated with the following process:

Initially for the corpus $\mathbb{D}$:

1. Choose a Dirichlet prior α describing the corpus.
2. Determine β, a $K \times |\mathbb{W}|$ matrix of word probabilities given topics, with $\beta_{jk} = P(w_j|z_t)$.
3. For each document $\mathbf{d} \in \mathbb{D}$ in the corpus:
   (a) Determine a document length $N \sim \text{Poisson}(\xi)$.
   (b) Choose $\theta \sim \text{Dir}(\alpha)$ as the parameter specifying the probability distribution of topics, given this document.
   (c) For each word place $\mathbf{n} \in [\mathbf{1}, \cdots \mathbf{N}]$ in the document:
      i. Choose a topic $z_n \sim \text{Discrete}(\theta)$.
      ii. Given the chosen topic z_n, choose a word w_n from the distribution $P(w_n|z_n, \beta)$.

Assuming the above process for document generation, LDA estimates the unknown α and β parameters from the corpus.[2] Using the estimations $\hat{\alpha}$ and $\hat{\beta}$, it is possible to infer any other parameter. The coupling between these two parameters, however, makes the problem intractable for a direct estimation [9]. Blei *et al.* therefore suggests a mean-field variational inference method as an approximation. This involves (1) introducing a set of variational parameters, γ and θ, (2) getting rid of α and β, and (3)

[2]Note that the Dirichlet distribution is chosen for its convenience of having finite dimensional sufficient statistics, and for being the conjugate prior to the discrete distribution, while the Poisson distribution for determining the length of documents is an arbitrary choice and not critical for the model. For more details, refer to [9].

Algorithm 1 Batch Gibbs sampling algorithm proposed by Griffiths and Steyvers [38]. Algorithm formulation taken from [14].

initialize $\vec{z} = [z_1, \cdots z_N]$ randomly from the set $\{1, 2, \cdots K\}$
while not converged **do**
  choose a word index j from $\{1, 2, \cdots N\}$
  sample z_j according to $P(z_j|\vec{z}_{\setminus j}, \vec{w}_N)$ (Equation 15)
end while

Algorithm 2 The Incremental-LDA method we propose for environments with unpredictable and dynamic context counts.

initialize context count $K \leftarrow 1$.
while there are new scenes to be encountered **do**
  run K-Incremental Gibbs sampler with K
  while there are words with confidence lower than threshold τ (*i.e.,* $\mathbb{C}_{low} \neq \emptyset$) **do**
    increment context count $K \leftarrow K + 1$
    run K-Incremental Gibbs sampler with K
  end while
end while

optimizing γ and θ in order to achieve the tightest lower bound on the new log likelihood. The problem then reduces to minimizing the Kullback-Liebler divergence (KL) between the original distribution and the introduced variational distribution:

$$(\gamma^*, \phi^*) = \underset{(\gamma,\phi)}{\operatorname{argmin}} \quad \mathrm{KL}\left(q(\theta, \vec{z}|\gamma, \phi) \;||\; p(\theta, \vec{z}|\vec{w}, \alpha, \beta)\right). \tag{14}$$

Blei *et al.* [9] describe an Expectation-Maximization method through which γ^* and ϕ^* can be estimated. However, other approximate solution strategies for LDA also exist, for instance see [38] for a collapsed Gibbs sampling solution, and [95] for a collapsed variational inference approach.

5.2 Extensions and Variants of LDA

Out of a variety of methods developed for topic analysis [87, 27, 45], LDA stands out for two features: First, it is a generative model. There exists other powerful, non-generative models for topic analysis (for instance, see [45]), however, being a generative model, LDA can assign probabilities to documents that have not been seen before. The second property is its allowing non-strict memberships of words to topics: A word may be generated by multiple topics, and according to which document it occurs in, considering the topic probability distribution of the document, a different topic might be assigned to the different occurrences of the word.

Algorithm 3 The K-Incremental Gibbs sampling approach we propose as a companion to Incremental-LDA.

initialize $\vec{z}_N$ from the previous solution for K contexts
initialize $z_t \leftarrow K + 1$, $\forall c_t \in \mathbb{C}_{low}$
reassign $z_{t'} \leftarrow K + 1$, with probability $\delta \ll 1$, if $c_{t'} \notin \mathbb{C}_{low}$
while not converged **do**
  choose a word index j from $\{1, 2, \cdots N\}$
  sample z_j according to $P(z_j|\vec{z}_{N\setminus j}, \vec{w}_N)$ (Equation 15)
end while

LDA has been extended in several directions: Griffiths and Steyvers [38] described a Gibbs sampling method for inference in LDA framework, which they used for discovering topics of abstracts from PNAS, successfully extracting 'hot topics' per years. Canini *et al.* [14] and Hoffman *et al.* [44] provided online versions for the originally batch algorithm, which enable working directly on an incoming document stream. Finally, Zhai and Boyd-Graber [107] relaxed the finite-vocabulary-size constraint.

A strong limitation of LDA is that it requires the number of topics to be fixed prior to any inference. In a typical clustering setting, it is common to deal with unknown number of clusters. LDA, on the other hand, does not have such a flexibility. This requirement to build a solution on assumed fixed settings is characteristic of the parametric approaches, where the parameters of the solution are defined a priori and do not change no matter how many training examples are encountered. Although parametric approaches are very widely used and successful in a variety of learning tasks (among well-known examples are regression, Fisher's discriminant analysis, Bayesian graphical methods, *etc.*), the necessity of predefining the parameters beforehand can be restrictive. In latent feature models case, different methods have been proposed for dealing with an unknown number of clusters, focusing specifically on Dirichlet-process and Bayesian solutions [4, 37, 28]. Targeting specifically the LDA problem, Teh *et al.* [94] proposed a Hierarchical Dirichlet Process framework which can start with infinitely many possible topics, and settle on the likeliest number of topics itself. Wang *et al.* [103] developed an online solution for this hierarchical setting.

5.3 An Incremental and Online Version: Incremental-LDA

Since a robot operates in a dynamic world, it needs to be able to discover newly emerging contexts with new interactions. To truly comply with developmental principles, the robot not only needs to estimate the ideal number of contexts, but also to validate its own prediction continuously and revise and update it if necessary; we cannot foresee this for it (for a very good discussion on what makes a system developmental, see [92]).

Since the previously proposed variations are either batch, parametrically dependent on the number of topics K, or trying to converge to an ideal K that is assumed constant over time, we enhance the original LDA methodology with a simple mechanism that allows both online learning, and dynamic updating of the ideal K value over time. This new variant, henceforth called Incremental-LDA, does not need the number of contexts to have been predefined, starting instead with the most general case of $K = 1$, and increasing the context count as necessary. We build this variation on the batch Gibbs sampling method for solving the standard LDA problem as proposed by Griffiths and Steyvers [38], detailed below.

Batch Gibbs Sampling Approach Introduced by Griffiths and Steyvers [38], the Batch Gibbs Sampling Approach assumes an alternative LDA model with an additional Dirichlet parameter ξ as a prior to ϕ:

$$\begin{aligned} w_i|z_i, \phi^{z_i} &\sim \text{Discrete}(\phi^{z_i}), \\ \phi &\sim \text{Dirichlet}(\xi), \\ z_i|\theta^{d_i} &\sim \text{Discrete}(\theta^{d_i}), \\ \theta &\sim \text{Dirichlet}(\alpha), \end{aligned}$$

This is called a "collapsed" method [14], because it integrates out the θ and ϕ parameters, and instead samples only the topic variables $\vec{z} = [z_1, \cdots z_N]$. It starts by randomly assigning $\vec{z}$, and then until convergence samples the topic assignment z_j for the word w_j in document d from the distribution of the instantaneous state:

$$P(z_j|\vec{z}_{\setminus j}, \vec{w}_N) \propto \frac{n^{w_j}_{z_j,\setminus j} + \xi}{n_{z_j,\setminus j} + |\mathbb{W}|\xi} \times \frac{n^{d}_{z_j,\setminus j} + \alpha}{n^{d}_{\cdot,\setminus j} + K\alpha}, \tag{15}$$

where $(.)_{\setminus j}$ notation stands for all items excluding the currently considered index j, therefore letting $\vec{z}_{\setminus j}$: the vector of all topics except z_j, $\vec{w}_N$: the vector of all words, $n^{w_j}_{z_j,\setminus j}$: the number of times that word w_j has been assigned to topic z_j, except at index j, $n_{z_j,\setminus j}$: the number of times that any word has been assigned to topic z_j, except at index j, $n^{d}_{z_j,\setminus j}$: the number of times that any word in document d has been assigned to topic z_j, $n^{d}_{\cdot,\setminus j}$: the total number of all words in document d, with $|W|$ denoting the size of the vocabulary set, and K denoting the topic count. The approach assumes symmetric Dirichlet

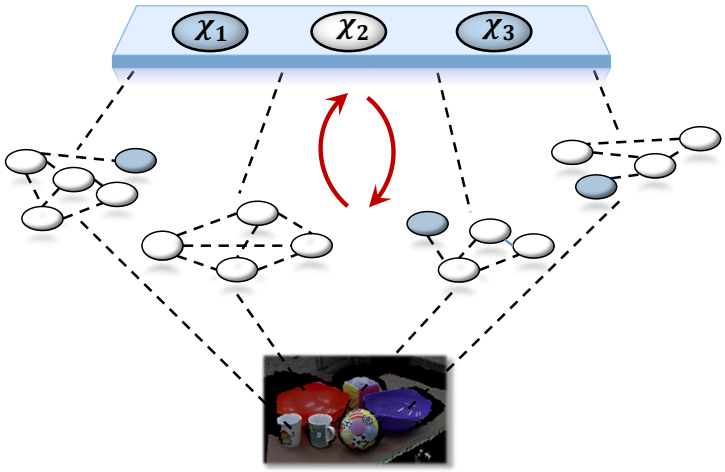


Figure 9: The feedback mechanism of contextual information to the concept web. Since an attentional mechanism is missing, we focus on the objects in the scene one by one, and extract separate concept webs for each of them. Then these concept webs are considered together for deciding on context. The contextual information is fed back to the concept web, and this feedback is iterated until convergence.

priors α and ξ, *i.e.*, that they are vectors with the same value in all entries. The α vs. ξ trade-off controls the compromise between the options of having few topics per document, vs. having few topics per word. The complete algorithm for the batch Gibbs sampling method is presented in Algorithm 1.

Incremental LDA Incremental-LDA instead decides on K dynamically, starting the with most general case, $K = 1$, and incrementing the context count until all the assignments can be made adequately. For making this decision, we define and use $\mathbb{C}_{low}$, the set of words whose topic assignment confidences are lower than a threshold value τ. If there exists such words with low confidences, *i.e.*, $\mathbb{C}_{low} \neq \emptyset$, Incremental-LDA attempts to increase their confidences by incrementing the context count. The complete procedure for Incremental-LDA is described in Algorithm 2.

K-Incremental Gibbs Sampling Incremental-LDA needs a modification of the batch Gibbs sampler, because this batch method starts from scratch each time a new context is incremented. The previous solution is forgotten completely, whereas parts of it would still be applicable, especially the parts with high enough confidence. Therefore, we introduce an incremental version of Gibbs Sampling: When the context count is incremented to $K + 1$, K-Incremental Gibbs Sampling resumes its search from the previously converged solution for K contexts, and conducts a local search in its close vicinity. This is done by retaining the previous assignments of the high-confidence terms, while initializing low-confidence terms ($\mathbb{C}_{low}$) to the newest context id $K+1$. Effectively, it reuses the highly confident part of the solution, instead of "reinventing" that part of the wheel. Note that for the sake of escaping possible local minima, a high-confidence term can also be reassigned to $K + 1$ with a low probability $\delta \ll 1$. The complete algorithm is depicted in Algorithm 3.

5.4 Using LDA to Model Context

In Section 4, we provide an explicit formalization of context. We propose LDA formulation is a particularly appropriate method for modeling this formalization, given its following properties:

- Due to the probabilistic nature of LDA, it allows non-strict assignment of words and documents to topics. For instance, if a certain word w occurs within the vicinity of group A of words in one kind of document, and group B of words in another, LDA can assign w to topic χ_A in the first case, and topic χ_B in the second case. This scenario corresponds to *consistent* concepts in our formalization, where w is consistent with both topics, with $w \in \mathbb{P}^{*A} \wedge w \in \mathbb{P}^{*B}$.
- If, on the other hand, a word w occurs within the vicinity of group C of words only, it is strongly

Table 5: The correspondence between the LDA terms and the notation used in this work.

LDA	Our Notation
document $d \in \mathbb{D}$	a single scene (the set of active concepts from the concept webs)
corpus $\mathbb{D}$	all scenes encountered during training phase
word $w_i \in \mathbb{W}$	an active concept c_{act} in the concept webs
	(can be a noun, adjective, or verb: $c_{act} \in \mathbb{C} = \mathbb{N} \cup \mathbb{A} \cup \mathbb{V}$)
topic	a 'context', either Kitchen, Playroom, or Workshop

associated with topic χ_C, such that its probability of belonging to other topics $\mathbb{Z} \setminus \chi_C$ diminishes to 0. This scenario corresponds to *enforcing* concepts in the formalization, with $w \in \mathbb{P}^{+C}$.

- If a word w never occurs within the vicinity of group D, its probability of belonging to D approaches to 0. In this case, w is a *conflicting* concept of topic χ_D, with $w \in \mathbb{P}^{-D}$.
- If two words $w_i \in \mathbb{P}^{+A}$ and $w_j \in \mathbb{P}^{*A}$ occur in a document together, due to the enforcing nature of w_i and consistent nature of w_j, LDA determines this document as of topic χ_A. w_i and w_j themselves are also associated with topic χ_A in this document.
- In contrast, if $w_i \in \mathbb{P}^{+A}$ and $w_m \in \mathbb{P}^{-A}$ occur in a document together, due to the conflict of the enforcing nature of w_i with conflicting nature of w_m, LDA assigns *two* topics to the document, both χ_A and $\chi_{\overline{A}}$, with $\overline{A} \neq A$, χ_A due to w_i and $\chi_{\overline{A}}$ due to w_m. This is a common scenario in real life, where items related to different topics can also be found together occasionally, which we called above a *mixed-context* scenario. In such a case, w_i is associated with topic χ_A and w_m is associated with topic $\chi_{\overline{A}}$.
- LDA works on the bag-of-words assumption that the order of the words in a document is not important, which is compatible with the unordered set formalization of context. Indeed, concepts exist or do not exist in a scene, there is no ordering between them. On the other hand, it does take into account the cardinality of concepts, the more instances of the same concept exists in a scene, the more strongly it affects the context.
- As detailed above, LDA can be made to operate in an online and incremental manner, which is consistent with our aim of lifelong development of robots in a changing world.

5.5 Extracting Context from a Scene

Context analysis is performed in these steps:

1. Each scene the robot encounters is represented as a set of active concepts
2. The sum of all the encountered scenes is then analogous to the corpus $\mathbb{D}$.
3. Each active concept c_{act} in this concept web is a word w_i in the document.
4. Finally, the "context"s that we are trying to discover are the topics of LDA.

Our aim is to associate each scene we encounter with its relevant contexts. Table 5 summarizes the correspondence between the LDA terms, and the notions in our robotics scenario.

In our framework, the robot follows an online learning scheme, in line with the *lifelong learning* principle of developmental robotics. Initially, it has zero knowledge. Similarly, at the beginning, it assumes that there is one general context, containing everything, with $K = 1$. As each scene is encountered, the objects in it are perceived and formed into a concept web, and the Incremental-LDA algorithm is called on this web. According to the confidence values assigned to the concepts and scenes, the context count is incremented if necessary.

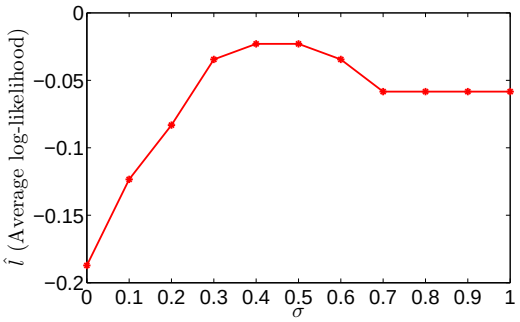


Figure 10: Average log likelihood $\hat{l}$ for varying σ. ($\sigma = 0$: Pure contextual information, $\sigma = 1$: Pure concept web decision, Equation 16). The interval $[0.4, 0.5]$ is depicted as maximizing $\hat{l}$. The convergence of $\hat{l}$ for $\sigma \geq 0.7$ corresponds to the contextual feedback being too weak to affect concept web decision at all, therefore the average log-likelihood does not vary in this region. We select $\sigma = 0.5$ from the interval $[0.4, 0.5]$.

5.6 Making Use of Context: Feeding the Contextual Information back to the Concept Web

Since the system does not employ an attentional mechanism currently, it focuses on each object in the scene one by one, finding the concepts of each object with MRF. The set of all these active concepts for all objects is then used for deducing the context of the scene. After determining the context, the probabilities of concepts are updated with the conditional likelihood of concepts in that context:

$$P(c)^* = \sigma \times P(\mathrm{c}) + (1 - \sigma) \times P(c|\mathcal{X}), \tag{16}$$

where $c \in \mathbb{C} = \mathbb{N} \cup \mathbb{A} \cup \mathbb{V}$ is a concept, $P(c)$ is the MRF-decided probability of the concept c, $\mathcal{X}$ is the context, $P(c|\mathcal{X})$ is the probability of the concept given the context (decided by Incremental-LDA), and $P(c)^*$ is the updated value of the concept probability. The concept web's MRF structure is then re-iterated until convergence. This iteration of (1) context deduction, (2) probabilistic update of concept web, and (3) reiteration of MRF loop is repeated until the contextual information and individual concept webs converge to a common value. A schematic representation of the system is presented in Figure 9.

Finally we try to designate a σ value for use in Equation 16, regulating the strength of contextual feedback in our world. A selection of $\sigma = 0$ would correspond to using only contextual information, whereas $\sigma = 1$ would consider only the concept web decision. An average log likelihood $\hat{l}$ is calculated over the test set as follows and depicted in Figure 10:

$$\hat{l} = \frac{1}{N|\mathbb{C}^{n+}|} \sum_{i=1}^{N} \sum_{c \in \mathbb{C}^{n+}} \log P(c|x_n, \sigma), \tag{17}$$

for varying σ, with N denoting the observation count, x_n denoting the n^{th} observation, $\mathbb{C}^{n+}$ denoting the set of concepts *related with* the n^{th} observation, with cardinality $|\mathbb{C}^{n+}|$, and $P(c|x_n, \sigma)$ denoting the probability of obtaining the related concept c given observation x_n, under the setting σ. The results estimate a reasonable interval between $[0.4, 0.5]$; from this interval, we select σ as 0.5.

5.7 Entropy-Based Evaluation of the System

We define an entropy-based metric of disorder to evaluate the performance of the system, combining two terms:

$$\tilde{H} = \rho \times H(C|X) + (1 - \rho) \times H(X|S), \tag{18}$$

where $H(.)$ is the entropy function, C, X, S are random variables denoting concepts, contexts, and scenes respectively, $H(C|X)$ is the conditional entropy of concepts given the context, $H(X|S)$ is the conditional entropy of contexts given the scene, and ρ is a parameter determining the relative importance of the two terms (set to 0.25 experimentally). These two terms stem from two possibly opposing targets: We

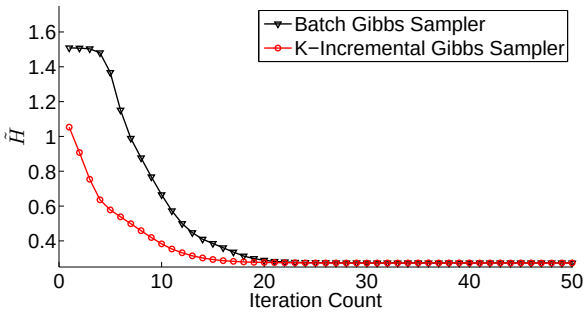


Figure 11: A comparison of the entropy ($\tilde{H}$) evolution (Equation 18) of the proposed K-Incremental Gibbs solver, versus the standard batch Gibbs solver. The K-Incremental Gibbs solver is fed a partial solution for $K = 2$ contexts, and then run for $K = 3$ contexts. The batch Gibbs sampler is directly run for $K = 3$ contexts. The K-Incremental Gibbs sampler is quicker to converge to the optimum solution in terms of entropy.

would like as few contexts (*topics*) as possible assigned to a scene (*document*), giving us more specific documents; and at the same time as few concepts (*words*) as possible associated with a context (*topic*). A combination of the two terms is expected to give us the most specific contextualization of the scene[3].

6 Experiments and Results

We now evaluate our framework and assumptions from three different aspects:

1. First, we investigate whether the proposed LDA variant, Incremental-LDA, performs well on determining the optimal number of contexts under insufficient number of scenes or contexts; *e.g.*, does it stop assigning new contexts at an optimal point. We also want to see if K-Incremental Gibbs sampler really provides better performance with fewer iterations, as expected, due to its reuse of partial solutions.

2. Then, we compare our approach against modeling context *directly* from raw features extracted from the scene. We evaluate if a densely-connected concept web is really beneficial in a context-discovery scenario.

3. Finally, we demonstrate how contextual information can improve reasoning. We present three different scenarios where context aids the robot with: (i) Scene evaluation and understanding, (ii) Object recognition, and (iii) Planning.

6.1 Performance of Incremental-LDA and K-Incremental Gibbs Sampling

First we evaluate the methods we introduced, Incremental-LDA and K-Incremental Gibbs Sampling. We try to gain insight on their efficiency, on their behavior under scarce, insufficient input data. We also seek to make sure that they can converge to the ideal context count, instead of trying to introduce new contexts forever. Therefore, in this section, we present basic sanity checks to understand how the system reacts in different stages of learning.

We first compare the performance of K-Incremental Gibbs sampling with that of batch Gibbs sampling. For justifying itself, the K-Incremental Gibbs sampler needs to converge faster, by reusing a previous partial solution, rather than starting from scratch. The test set we use includes 100 scenes of our robotic scenario, composed of 3 contexts (*Kitchen*, *Playroom*, and *Workshop*), over the 6 noun and 10 adjective concepts mentioned above. For generating this set, a context is decided randomly for each scene, and then the scene is populated with randomly chosen objects from the selected context. The concept webs corresponding to the objects are built, and the set of all these active concepts is used to describe the scene. Figure 11 presents the results over this test set that conform with our expectations:

[3]Similar multi-objective optimization of these two metrics can be found in the literature, for instance see [38].

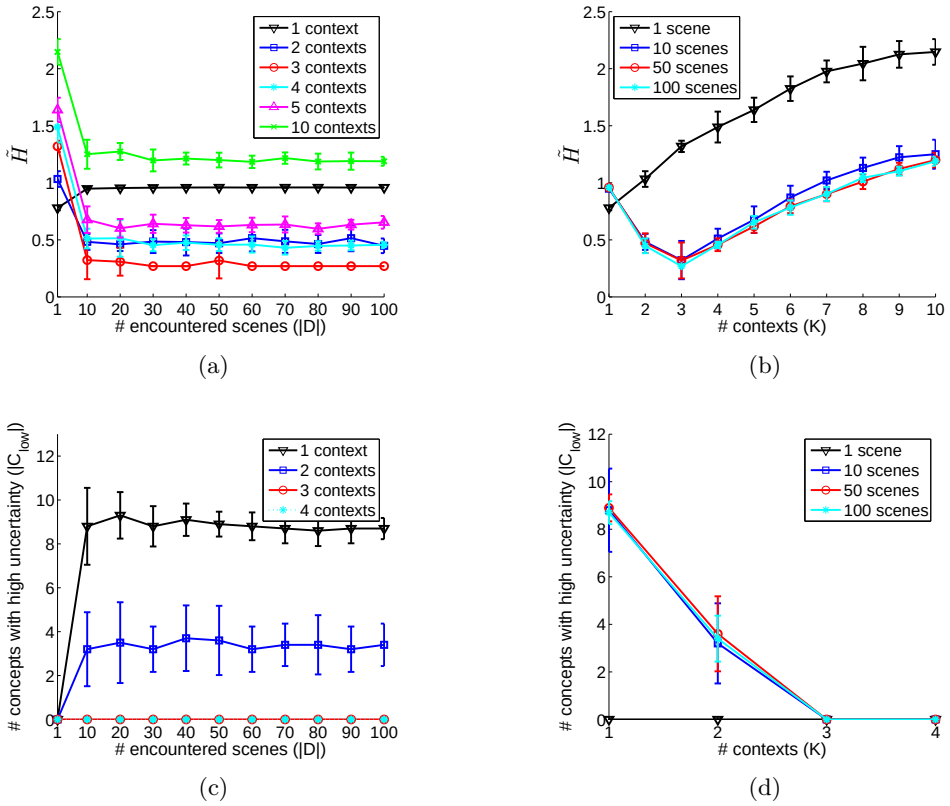


Figure 12: The effect of encountered scene counts and increasing numbers of context count K. Note that Incremental-LDA would itself stop at $K = 3$, however we forcefully continue increasing K for the sake of comparison. (a) Effect of encountered scenes on the entropy of the system, $\tilde{H}$, for different context counts. It is seen that the system converges in virtually all cases when it has encountered 10 scenes. (b) Effect of increasing K on the entropy of the system, $\tilde{H}$, for increasing number of scenes. The lowest entropy values are observed for $K = 3$ contexts, in line with the three contexts of our setting. (c) Effect of encountered scenes on the uncertain concepts, $|\mathbb{C}_{low}|$, for different context counts. $|\mathbb{C}_{low}|$ also converges by the time 10 scenes has been encountered. (d) Effect of increasing K on the uncertain concepts, $|\mathbb{C}_{low}|$, for increasing number of scenes. By $K = 3$ contexts $|\mathbb{C}_{low}|$ diminishes to 0, therefore Incremental-LDA would stop adding new contexts at this point. In all the experiments, 10 test sets of $|\mathbb{D}|$ scenes each are used. The mean values for the 10 test sets are plotted, while the standard deviations are indicated with error bars. [Best viewed in color]

When run for these $K+1 = 3$ contexts, K-Incremental Gibbs sampler (using a partial solution for $K = 2$) does converge faster, compared to the batch solver. We measure the convergence of the system in terms of its entropy (the entropy value eventually reached by the two solvers is indeed the expected minimum entropy value for these environmental conditions).

Next, we analyze Incremental-LDA under two variables: One is an increasing number of encountered scenes, in which case we hope to achieve convergence to a sane result as early as possible, and the second is the case of a varying number of contexts K, in which we look for a preference for the expected number of contexts, $K = 3$. We use the measure of entropy, $\tilde{H}$ (Equation 18), as well as the (possible) decrease in the number of concepts with high uncertainty ($\mathbb{C}_{low}$), as our performance metrics. Note that, left alone, Incremental-LDA would itself converge to a certain K setting, which is ideally $K = 3$ here, however, for the sake of comparison, we forcefully set varying K values in these experiments.

We use 10 test sets of $|\mathbb{D}|$ scenes in each case, obtained randomly from the 3 contexts. For generating each scene, a context is again selected randomly, and randomly chosen objects from the selected context are populated into the scene. The number of total scenes in a test set, $|\mathbb{D}|$, is varied according to the experiment conditions. Figure 12(a) presents the time evolution of the entropy of the system, $\tilde{H}$, as

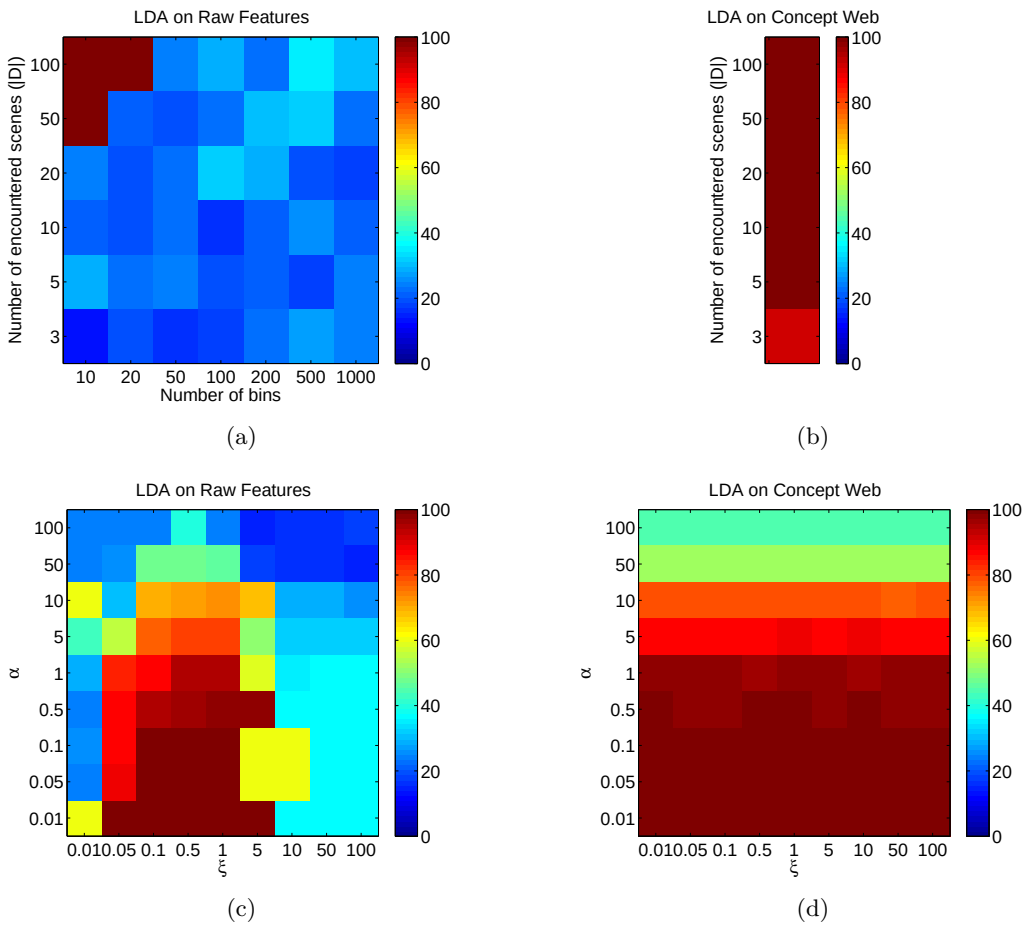


Figure 13: Comparisons of LDA over raw features only, with LDA over MRF-based concept web. Presented values are the predicted likelihood of "correct contexts" in each corresponding case. For evaluation, the correct contexts have been extracted from ground truth through supervision. (a-b) Predicted likelihoods of correct contexts for varying numbers of encountered scenes, and for varying amounts of discretization in raw-features case (*i.e.*, "bin count"). Parameter settings of $\alpha = 0.1$ and $\xi = 0.1$ have been used. (a) Results using only the raw features as input to LDA, discretized to depicted bin counts. (b) Results using the concept web as the input. Due to no discretization to bins being necessary, the results are 1-dimensional. (c-d) Predicted likelihoods of correct contexts for varying settings of α and ξ, in the raw-features only (c) and concept web (d) cases. 50 scenes and 10 bins (for the raw features case) have been used. [Best viewed in color]

more scenes are being encountered, under different hard-coded settings of K. The mean and standard deviation values for the 10 test sets are indicated with error bars. In this setting of increasing $|\mathbb{D}|$, we wish to achieve as early convergence as possible, which is duly achieved by the 10 scenes mark. This shows that the system is indeed able to converge, and it converges rapidly. From a different point of view, Figure 12(b) shows how the entropy of the system would change if different K values were used. The results show that, for reasonable numbers of scenes, the lowest possible entropy values are achieved when $K = 3$, which conforms our expectations since, in our experiments, we have three contexts: *Kitchen*, *Playroom*, and *Workshop*.

Then, we evaluate the number of concepts with high uncertainty, $\mathbb{C}_{low}$, in these two cases. Incremental-LDA, as mentioned above, aims to minimize this number. Figure 12(c) shows that this number also converges when 10 scenes are encountered, diminishing as well for more than three contexts. For a lower context count, it again stabilizes to a certain positive number. Figure 12(d) displays the same results from a varying context count point of view.

Table 6: Prediction of context for a few example scenes where confidences are indicated in parentheses. Bold text indicates correct decisions whereas stroked text indicates wrong decisions. [Best viewed in color]

Pure Contexts			**Mixed Contexts**		
Scene	**Existing Objects**	**Predicted Context (% contribution)**	**Scene**	**Existing Objects**	**Predicted Context (% contribution)**
	2 cups,2 plates	**Kitchen (100%)**		3 boxes, 1 ball 1 cylinder, 1 tool	**Playroom (72, 59%)** **Workshop (26, 23%)** Kitchen (1.18%)
	2 boxes, 2 balls	**Playroom (100%)**		2 plates, 2 cup, 1 ball, 1 box	**Kitchen (62.04%)** **Playroom (37.14%)** Workshop (0.82%)
	2 tools,3 cylinders	**Workshop (100%)**		1 tool, 1 cylinder, 1 plate, 1 cup	**Kitchen (46.67%)** **Workshop (51.56%)** Playroom (1.77%)

In all four cases, the system converges at about 10 scenes, and shows preference (in terms of minimal entropy and minimal number of highly uncertain concepts) at $K = 3$ contexts. Since this is also the point at which $|\mathbb{C}_{low}|$ reaches to zero, Incremental-LDA stops adding new counts at this point, which indeed corresponds to the minimum entropy setting of our system.

6.2 Context from the Concept Web against Context from Raw Features

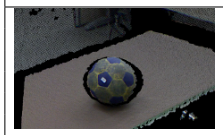
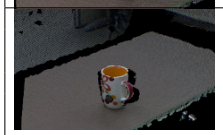
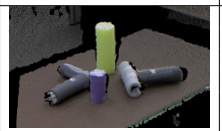
Next we evaluate how useful the concept web is in guiding contextualization. Figure 13 shows the comparison of LDA on concept web versus LDA on raw-features-only. First, we contrast how the two schemes fare in case of insufficient scene encounters. Concurrently, we also investigate to what degree the discretization of the raw-features is necessary, if at all. In the second type of tests, we conduct a grid parameter search in the LDA space, to decide what are the best parameters to run the both algorithms over, and how much they are sensitive to parameter changes. Note that these two sets of experiments must be thought of in unison, in the sense that we have iteratively updated the parameters used in one set according to the best results of the other set, therefore we hope to present meaningful results in both sets. In the figures, we present the predicted likelihoods assigned by these algorithms to the contexts that we "know" to be true. The correct contexts have been decided through supervision for evaluation purposes only.

Figure 13(a) versus Figure 13(b) depicts the results of the first set, *i.e.*, the effects of scene count and discretization (with $\alpha = 0.1$, $\xi = 0.1$) An important result that pops out is that the raw-features approach needs 50 scenes to settle on a meaningful partitioning, while the concept web method manages to converge with an impressive speed at as few as 5 scenes. Even at 50 scenes, the raw-features approach needs to be supported by coarse discretization of the features (*i.e.*, being divided into 10 bins at most), since LDA is unable to locate statistically significant co-occurrences otherwise. For other settings, the decisions of the raw-features approach are at chance level: 33.3% for a 3-way decision.

Figures 13(c) and 13(d), on the other hand, present the results of the grid search in the α-ξ space (with 50 scenes, 10-bins of discretization). Once again, we see that LDA-on-raw-features is more fragile against parameter changes, while the concept web method proves robust under most settings. Indeed, even for the worst parameter settings, notice that the concept-web case provides confidences of over 50%, which are sufficient for correct decision making, and are well over the chance level of 33.3%.

The results confirm that learning context from concepts is better than learning them from raw features in two aspects: (i) Learning converges faster, and is therefore more reliable even after as few as 3-4 scene encounters, and (ii) It is less sensitive to the model parameters, which increases the robustness of learning without needing a careful tuning of parameters.

Table 7: Object recognition in context. Prediction confidences are indicated in parentheses. Bold text indicates correct decisions whereas stroked text indicates wrong decisions. [Best viewed in color]

	Perception only			Concept Web			In Context			
Objects	Predicted Nouns (% confidence)	Predicted Adjectives (% confidence)		Predicted Nouns (% confidence)	Predicted Adjectives (% confidence)		Context	Predicted Nouns (% confidence)	Predicted Adjectives (% confidence)	
	ball (8%) box (13%) **cup (43%)** cylinder (20%) plate (9%) tool (7%)	edgy (34%) **hard (71%)** noisy (42%) **short (54%)** thick (47%)	**round (66%)** soft (29%) **silent(58%)** tall (46%) ~~thin (53 %)~~	ball (0%) box (0%) **cup (100%)** cylinder (0%) plate (0%) tool (0%)	edgy (0%) **hard (100%)** noisy (0%) **short (100%)** **thick (100%)**	**round (100%)** soft (0%) **silent(100%)** tall (0%) thin (0 %)		ball (0%) box (0%) **cup (100%)** cylinder (0%) plate (0%) tool (0%)	edgy (0%) **hard (100%)** noisy (0%) **short (100%)** **thick (100%)**	**round (100%)** soft (0%) **silent(100%)** tall (0%) thin (0 %)
	**ball (33%)** box (16%) cup (13%) cylinder (13%) plate (14%) tool (11%)	edgy (42%) hard (39%) **noisy (62%)** **short (61%)** **thick (56%)**	**round (58%)** **soft (61%)** silent(38%) tall (39%) thin (44 %)	**ball (100%)** box (0%) cup (0%) cylinder (0%) plate (0%) tool (0%)	edgy (0%) hard (0%) **noisy (100%)** **short (100%)** **thick (100%)**	**round (100%)** **soft (100%)** silent(0%) tall (0%) thin (0 %)		**ball (100%)** box (0%) cup (0%) cylinder (0%) plate (0%) tool (0%)	edgy (0%) hard (0%) noisy (0%) **short (100%)** **thick (100%)**	**round (100%)** **soft (100%)** **silent(100%)** tall (0%) thin (0 %)
	ball (12%) box (13%) cup (17%) **cylinder (29%)** plate (12%) tool (17%)	edgy (45%) **hard (56%)** **noisy (58%)** short (41%) thick (40%)	**round (55%)** soft (44%) silent(42%) **tall (59%)** ~~thin (60 %)~~	ball (0%) box (0%) cup (0%) **cylinder (100%)** plate (0%) tool (0%)	edgy (0%) **hard (100%)** **noisy (100%)** short (0%) thick (0%)	**round (100%)** soft (0%) silent(0%) **tall (100%)** ~~thin (100 %)~~		ball (0%) box (0%) cup (0%) **cylinder (100%)** plate (0%) tool (0%)	edgy (0%) **hard (100%)** **noisy (100%)** short (0%) **thick (100%)**	**round (100%)** soft (0%) silent(0%) **tall (100%)** thin (0 %)
	ball (14%) **box (43%)** cup (12%) cylinder (12%) plate (11%) tool (8%)	**edgy (64%)** hard (34%) noisy (30%) **short (59%)** **thick (63%)**	round (36%) **soft (66%)** **silent(70%)** tall (41%) thin (37 %)	ball (0%) **box (100%)** cup (0%) cylinder (0%) plate (0%) tool (0%)	**edgy (100%)** hard (0%) noisy (0%) **short (100%)** **thick (100%)**	round (0%) **soft (100%)** **silent(100%)** tall (0%) thin (0 %)		ball (0%) **box (100%)** cup (0%) cylinder (0%) plate (0%) tool (0%)	**edgy (100%)** hard (0%) noisy (0%) **short (100%)** **thick (100%)**	round (0%) **soft (100%)** **silent(100%)** tall (0%) thin (0 %)
	ball (12%) box (13%) cup (12%) cylinder (14%) **plate (40%)** tool (9%)	edgy (46%) **hard (53%)** noisy (44%) short (45%) **thick (59%)**	**round (54%)** soft (47%) **silent(56%)** tall (55%) thin (41%)	ball (0%) box (0%) cup (0%) cylinder (0%) **plate (100%)** tool (0%)	edgy (0%) **hard (100%)** noisy (0%) short (0%) **thick (100%)**	**round (100%)** soft (0%) **silent(100%)** **tall (100%)** thin (0 %)		ball (0%) box (0%) cup (0%) cylinder (0%) **plate (100%)** tool (0%)	edgy (0%) **hard (100%)** noisy (0%) short (0%) **thick (100%)**	**round (100%)** soft (0%) **silent(100%)** **tall (100%)** thin (0 %)
	ball (11%) box (13%) cup (15%) cylinder (18%) plate (11%) **tool (32%)**	edgy (48%) **hard (55%)** **noisy (61%)** short (39%) **thick (57%)**	**round (52%)** soft (45%) silent(39%) **tall (61%)** thin (43 %)	ball (0%) box (0%) cup (0%) cylinder (0%) plate (0%) **tool (100%)**	edgy (0%) **hard (100%)** **noisy (100%)** short (0%) **thick (100%)**	**round (100%)** soft (0%) silent(0%) **tall (100%)** thin (0 %)		ball (0%) box (0%) cup (0%) cylinder (0%) plate (0%) **tool (100%)**	edgy (0%) **hard (100%)** **noisy (100%)** short (0%) **thick (100%)**	**round (100%)** soft (0%) silent(0%) **tall (100%)** thin (0 %)
	ball (14%) box (17%) cup (19%) **cylinder (26%)** plate (13%) tool (11%)	edgy (42%) **hard (60%)** noisy (42%) short (45%) thick (40%)	**round (58%)** soft (40%) **silent(58%)** ~~tall (55%)~~ **thin (60 %)**	ball (0%) box (0%) ~~cup (100%)~~ cylinder (0%) plate (0%) tool (0%)	edgy (0%) **hard (100%)** noisy (0%) **short (100%)** thick (0%)	**round (100%)** soft (0%) **silent(100%)** tall (0%) **thin (100 %)**		ball (0%) box (0%) cup (0%) **cylinder (100%)** plate (0%) tool (0%)	edgy (0%) **hard (100%)** noisy (0%) **short (100%)** thick (0%)	**round (100%)** soft (0%) **silent(100%)** tall (0%) **thin (100 %)**

6.3 Using Context, Part 1: Making Sense of Pure- and Mixed-Context Environments

Now we demonstrate how our context model can be utilized in reasoning and decision making. The first scenario is designed for assessing how successful our model is in recognizing contexts of scenes. The robot encounters six different scenes, three of which are composed of items of a single context, and the remaining three of multiple contexts. Table 6 demonstrates the predicted context(s), showing that the robot can distinguish between pure and mixed-context scenes correctly, and decide on the correct components in case of a mixed-context scene. These results are important, because they demonstrate that our interpretation of the scene context is correct, regardless of the scene being composed of a single context or multiple contexts. Therefore, we obtain justification for our next step of using this contextual interpretation for guiding reasoning in other cognitive tasks.

6.4 Using Context, Part 2: Object Recognition in Context

The second scenario we consider studies the effect of context on object recognition. Table 7 demonstrates the recognition results for seven sample objects that are either (i) individually perceived (columns 2-3), (ii) assessed in an individual concept web (columns 4-5), or (iii) evaluated in context (columns 7-8).

The results show that concept web itself can correct certain mistakes of the perception-only assessment, while also boosting confidences of guesses to 100% certainty. However, it is not flawless and is also prone, albeit in a lesser amount, to errors (see the 3^{rd} and 7^{th} rows in the table). In such cases, it is especially difficult to correct these errors, due to the initial high confidence associated with them. Contextual information can be beneficial in these settings: Remembering our fundamental assumption that related objects occur together in context (which allowed us to develop an LDA-based model in the first place), the system can use context to revise and correct its previous judgments. The loop of (a) context deduction, (b) probabilistic update of concept web, and (c) reiteration of MRF, as described in Section 5.6 and Equation 16, is utilized for refining predictions in context. Combined results for all 15 objects in our test set are demonstrated in Figure 14, which also show an improvement of performance for the context-guided recognition.

6.5 Using Context, Part 3: Planning in Context

Finally, we show how contextual information can be useful in a planning task. It is known that humans hugely rely on contextual information for planning their actions [91], possibly due to a severely restricted working memory capacity [23, 31], which results in efficient day-to-day planning, but maybe less-than-favorable performances in chess. The robots would also benefit from similar contextual guidance in planning.

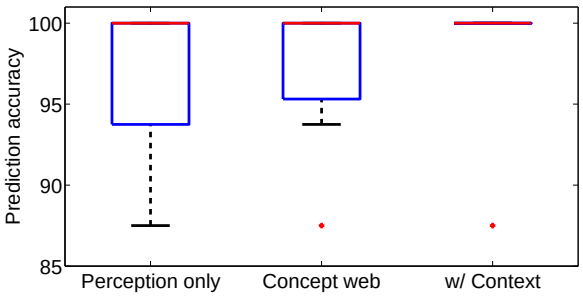


Figure 14: The combined results of object recognition in context, over all 15 objects in the test set. The prediction accuracies over all determined noun and adjective concepts, using (1) only perceptual features, (2) the concept web, and (3) contextual information are compared. In the plot, the red lines denote the median values, the boxes denote the data that fall between the 25^{th} and 75^{th} percentiles, the whiskers cover the extreme data that are not outliers, and stars indicate the outliers.

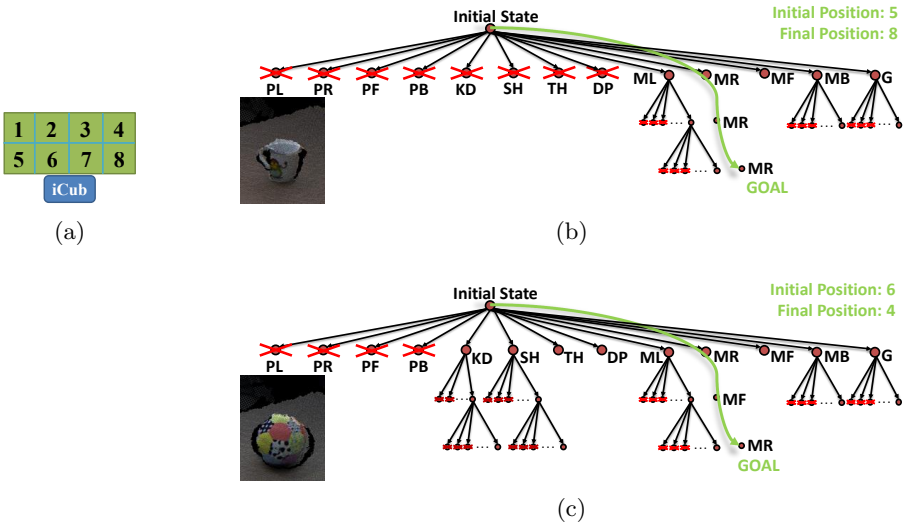


Figure 15: Pruning of forward planning trees by integrating contextual information. (a) iCub's workspace schematized. (b-c) Two planning scenarios. The branches that are pruned due to being irrelevant for the current context are shown with crosses. The behavior abbreviations stand for: PL: Push left, PR: Push right, PF: Push forward, PB: Push backward, KD: Knock down, SH: Shake, TH: Throw, DP: Drop, ML: Move left, MR: Move right, MF: Move forward, MB: Move backward, G: Grasp. (b) First planning scenario. iCub is expected to move a cup from position 5 to position 8. Since pushing and knocking actions are dangerous in the kitchen context, these nodes are pruned without further expansion. (c) Second scenario. iCub must bring a ball from position 6 to 4. This time pushing are pruned, since pushing a ball causes it to roll down from the table.

To show how context can be used similarly in a robotic planning scenario, we provide two simple situations as proof-of-concept: The robot has to move two objects over a table (Figure 15(a)) from an initial to a goal position. Since the robot has learned the effect features of behaviors on training objects, it is theoretically able to expand a planning tree starting from the initial state and expanding behavior nodes until the goal condition is reached.

In the first scenario, Figure 15(b), the robot is asked to move a cup from position 5 to position 8. This goal can be achieved with three consecutive *move right* actions in our setting. A fully-expanded tree, therefore, would consist of three levels, and with a branching factor of 13, it will consist of $13^0 + 13^1 + 13^2 + 13^3 = 2380$ nodes. However, given the contextual information of the scene, which is the *Kitchen* context, the robot can refrain from expanding the inappropriate behaviors in a Kitchen[4], leaving only the *move left*, *move right*, *move forward*, *move backward* and *grasp* as possible actions to be expanded. Such an elimination gives a drastic reduction in the size of the planning tree, resulting in $5^0 + 5^1 + 5^2 + 5^3 = 156$ nodes instead of 2380.

Figure 15(c) shows another scenario in the *Toy* context. This time, the robot refrains from applying the *push* actions on associated objects, since balls, which are also in this context, tend to roll down and fall from the table when pushed. Therefore, the *push* nodes are pruned, leaving $9^0 + 9^1 + 9^2 + 9^3 = 820$ nodes in the tree. We use a breadth-first forward planning scheme subject to context-dependent pruning, as depicted in Algorithm 4.

Figure 16 compares un-pruned and pruned node counts for 10000 random scenarios in the move-over-the-table scenario presented above, presented for the three contexts separately. Each scenario is prepared by randomly determining a context, as well as initial and goal positions on the table environment, and then asking the robot to plan a behavior sequence from the initial to the goal position in this contextual background.

The reductions shown here are only provided as proof-of-concepts, but it is clear how important it

[4]Assuming we do not want to, for instance, *shake* a full cup.

Algorithm 4 Breadth-first forward planning with context-dependent pruning.

if goal position p_g = initial position p_i **then**
  return empty plan []
end if
QUEUE $\leftarrow$ [[b_1], ... , [b_I]], $\forall b_i \in \mathcal{B}_\mathcal{A}$, $\mathcal{B}_\mathcal{A}$: the set of applicable behaviors in the current context
while QUEUE is not empty **do**
  pop PLAN from QUEUE
  - Predict the outcome of the behaviors in the PLAN:
  current position $p_c \leftarrow$ initial position p_i
  for all behavior b_i in PLAN **do**
    update current position: $p_c \leftarrow b_i[p_c]$
  end for
  - Check whether we have reached the goal:
  if current position p_c = goal position p_g **then**
    return PLAN
  end if
  - Add possible behaviors in the current context as alternative plans:
  for all behavior $b_j \in \mathcal{B}_\mathcal{A}$ **do**
    if next position due to b_j ($p_n \leftarrow b_j[p_c]$) is within table boundaries **then**
      push PLAN.append([b_j]) to QUEUE
    end if
  end for
end while
return empty plan []

is for a robot to learn to prune its search trees in a real world setting. For a very limited robot of a small, or maybe even intermediate set of actions, considering each action for every situation might be an option, but for any robot who aims to operate in the real world, the actions will be so varied and planning chains will necessarily be so long that even most basic reductions (*i.e.*, no need to consider opening the kitchen door for heating a glass of milk) will be of critical importance.

7 Summary and Discussion

In the article, we studied how a humanoid robot can model, learn and use context. For modeling context, we employed and extended Latent Dirichlet Allocation (LDA), a widely-used topic model in

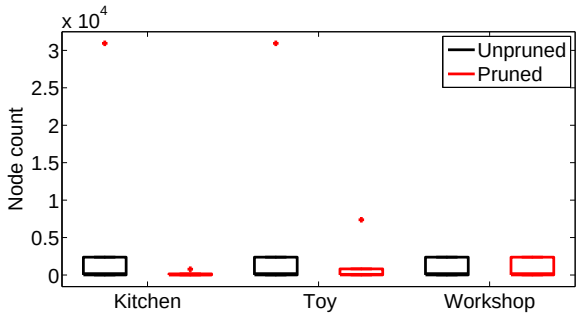


Figure 16: The node counts of *unpruned* vs. *pruned* planning trees of 10000 random scenarios, grouped by their contexts. The Kitchen context is subject to more pruning, as expected, due to a large number of *NA* behaviors. The Workshop context, on the other hand, is not subject to any pruning, since all behaviors are potentially applicable. In the plot, the boxes denote the data that fall between the 25^{th} and 75^{th} percentiles, and stars indicate the outliers. [Best viewed in color]

the computational linguistics literature. Unlike the existing applications of LDA in robotics for, *e.g.*, word learning, where LDA is directly applied onto low-level sensorimotor data, we were motivated by the concept web hypotheses in humans and its computational advantages to apply LDA onto a concept web model that we developed in our previous work using Markov Random Fields.

We demonstrated the following important aspects:

- In an unsupervised fashion, the robot can learn context even if the number of contexts is not given. By using an online version of the Gibbs sampler proposed in the article, the robot can work online to process new observations and can tackle new contexts. By a systematic analysis, we show that the model finds the *correct* number of contexts in different settings.
- The robot can use the learned contexts to improve its performance in cognitive tasks. In the article, we showed this aspect for object recognition and planning.
- Finally we show how learning context over a web of abstracted concepts is *easier* and provides better performance for an LDA-based architecture, which deals with the sensorimotor complexity of real world better than raw features themselves.

Below, we discuss several aspects of our design.

7.1 Basing Context on the Concept Web

Biological Plausibility We based our context model on a concept-based framework where concepts of different types are connected to each other in a web structure. Our motivation comes from the hypothesis that human cognition is mostly based on concepts that are connected in a web. This idea has support from different perspectives, with evidence demonstrating the tight connection of concepts to motor [80, 79] and sensory cortices [34, 50], and "complex" concepts being lost in semantic dementia, with more generic concepts staying intact [56, 57, 76, 62]. It should be noted that such a formation would also make sense from a purely computational point of view, with the distribution of concept symbols over the brain making them more low-cost to implement: Steel's Recruitment of Language theory [90] pointed out how language might have "piggy-backed" on other well-developed parts of the brain in order to develop symbolic representation. Damasio [25] also envisioned memory and consciousness as a time-synchronized activation of multiple areas of cortex in a controlled manner. Recently, Mitchell *et al.* [67] showed how cortical activity for a complex concept can be "predicted" in terms a superposition of the activities for its "basis" concepts, whereas Huth *et al.* [46] proposed how a continuous semantic space may be hosting thousands of concepts in humans, with more related concepts being kept close, and unrelated ones apart, in a very high dimensional but still meaningful cortical grid. Therefore, evidence supports a concept web hypothesis, which makes the concepts robust, easily accessible, and low-cost to 'implement' biologically.

Computational Advantages In addition to the biological support, basing context on a concept web has computational advantages: In [15], we show how concept web enables a superior performance of object recognition and conceptualization as compared to a raw-feature based scheme. In this work, we provide further evidence regarding the performance of concept web for LDA-based conceptualization. We demonstrate how the concept web provides better performance with significantly fewer training examples, as well as reduced sensitivity against system parameters. These advantages are due to its abstraction capability: The real world presents an overwhelming amount of complex information, which needs some structure to be imposed before statistically significant relations can be discovered. This is argued to be the driving reason of conceptualization in humans as well (*e.g.*, [73, 39, 51, 26, 53, 96, 40], for a slightly different but interesting argument, see also [41].)

7.2 Planning in the Real World

Bylander [13] and Chapman [18] show that planning is intractable in the general sense, unless it is restricted severely, for instance, to propositional planning with strictly positive preconditions and exactly one postcondition. Such restricted cases can be defined to reduce the planning problem to a polynomial-time subset; however, small deviations make the problem intractable again: *e.g.*, the NP-hard problem

of allowing two postconditions along with one precondition, or the NP-complete problem of one strictly positive postcondition along with one precondition. As Bylander [13] and Hendler [42] note, it is difficult to describe any interesting world in propositional logic, let alone such restrictions for the sake of tractability. We have to find a workaround. We propose that this workaround can be, and for humans, is context [59, 101, 24, 29].

7.3 Limitations and Future Work

Overall, we provide promising results that a learning scheme which *includes* background information, instead of leaving it out, is feasible *and* useful for a robot when dealing with the real world. Our work can be extended in several directions.

The experiments were performed on real objects, although the settings are not realistic. This limitation was due to the interaction capabilities of iCub: iCub cannot walk and is confined to a table-top environment. Moreover, due to its delicate hands and the limited precision of the touch sensors on the hands, the range of objects that can be interacted with was limited to light-weight and convex objects. This also restricted us in the varieties of contexts. However, we claim that our model is applicable to more realistic settings, with more objects and contexts, and successful applications of LDA in complicated, challenging linguistic settings are promising indications of that.

It should also be noted that, although our current concept web is comprised of noun, adjective, and verb concepts, a cognitive model should include spatial, temporal, adverb, and social concepts as well. With the incorporation of these types of concepts in our concept web, contexts related to their semantics will also be able to manifest themselves in our model.

Another plausible extension is regarding the concept web: The current concept web is a model of long-term memory only, with links holding information about the robot's experiences about the world. This long-term memory is activated based on the current perception, yet, there is no clear separation between short-term and long-term memory akin to humans.

Acknowledgments

We would like to thank Angelo Cangelosi, Anna Borghi and Honghai Liu for fruitful discussions on integrating context into cognitive systems. For the experiments, we acknowledge the use of the facilities provided by the the Modeling and Simulation Center of METU (MODSIMMER). This work is funded by the Scientific and Technological Research Council of Turkey (TÜBİTAK) through project no 111E287.

A Appendix: Prototype Extraction and Context Assignment

We use prototypes to represent the noun ($\mathbb{N}$), adjective ($\mathbb{A}$) and verb ($\mathbb{V}$) concepts. The noun and adjective concepts are related to the object entities, while verb concepts are related to the changes induced on the objects by the behaviors. Therefore, the prototypes of the noun and adjective concepts are obtained from the entity feature vectors $\mathbf{e}$, while the verb concept prototypes are obtained from effect feature vectors $\mathbf{f}$. Each object in the training set is labeled beforehand by supervision to denote the concepts it is associated with: Each training object is strictly labeled with 1 noun concept (out of 6) and 5 adjective concepts (one from each of the 5 dichotomic pairs). In addition, every applicable behavior is applied to each training object, and the interactions are labeled with strictly 1 verb concept.

During training, the entity and effect feature vectors are collected from the training objects, and divided according to the labeled concepts. For each concept, every feature is assessed in terms of its contribution to the concept: If the feature has a highly positive contribution to the concept, it is indicated with a '+' in the concept prototype. '-' denotes a negative contribution, and '*' denotes inconsistent contribution. These contributions are decided by clustering the features, using Robust Growing Neural Gas (RGNG) clustering algorithm [81], in a two dimensional space of means and variances: The mean axis denotes the amount of the contribution, while the variance axis denotes the consistency. Features with positive mean and low variance are marked with '+'; negative mean and low variance with '-'; and high variance with '*'. Of special interest are the features marked with '*'s, which effectively distinguishes *irrelevant* features, that can be disregarded from comparisons regarding the concept.

Algorithm 5 Derivation of a prototype from the exemplars of a concept (Adapted from [48])

for all c in the set of concepts $\mathcal{C}$ **do**
- Compute the mean μ_c^i for each feature dimension i:

$$\mu_c^i = \frac{1}{|\mathbb{E}(c)|} \sum_{e \in \mathbb{E}(c)} e^i, \tag{19}$$

where $\mathbb{E}(c)$ denotes the set of examplars of concept c, $|\mathbb{E}(c)|$ is the cardinality of the set $\mathbb{E}(c)$; and $\mathbf{e^i}$ is the i^{th} value of vector $\mathbf{e}$.
- Compute the variance σ_c^i of each feature dimension i:

$$\sigma_c^i = \frac{1}{|\mathbb{E}(c)|} \sum_{e \in \mathbb{E}(c)} (e^i - \mu_c^i)^2. \tag{20}$$

end for

- Apply Robust Neural Growing Gas (RGNG) clustering algorithm in the space of $\mu \times \sigma$ of the features.
for all c in the set of concepts $\mathcal{C}$ **do**
  if $c \in \mathcal{N} \cup \mathcal{A}$ **then**
    - Manually assign the labels '+', '-', and '*' to the three clusters that emerge in the previous step, according to:
    if cluster is high on μ axis and low on σ axis **then**
      label with '+'
    else if cluster is low on both axes **then**
      label with '-'
    else if cluster is high on σ axis **then**
      label with '*'
    end if
  else
    - Manually assign the labels '+', '-', '*', and '0' to the four clusters that emerge in the previous step, according to:
    if cluster is high on μ axis and low on σ axis **then**
      label with '+'
    else if cluster is low on both axes **then**
      label with '-'
    else if cluster is located around 0 on μ axis and is low on σ axis **then**
      label with '0'
    else if cluster is high on σ axis **then**
      label with '*'
    end if
  end if
end for

Prototypes for the verb concepts are extracted in a similar manner, except that (1) they are calculated over the effect features **f**, and (2) they include a '0' character for features that are unaffected by the behavior.

Eventually, we obtain 29 prototypes in total; 6 for nouns, 10 for adjectives, and 13 for verbs. The prototypes of the noun and adjective concepts are of length 91, the same with the length of an entity feature vector **e**, containing 66 visual, 13 audio, 6 haptic and 6 proprioceptive features. The prototypes of the verb concepts are composed of 66 characters, and denote visual features only. The prototypes used in this study are shown in Table 8. The complete procedure is depicted in Algorithm 5.

When a new object is encountered, its entity feature vector **e** is compared against the noun and adjective prototypes. Similarly, if a behavior has been applied, the effect feature vector **f** is compared against the verb concept prototypes to recognize the behavior. This comparison consists of finding the concepts that minimize the Euclidean distance between the object's feature vector and the concept mean vector (Equation 1). The *irrelevant features* of each concept, marked with '*' in the concept prototype, are excluded from this calculation.

Table 8: Extracted prototypes for noun, adjective and verb concepts (Taken from [15])

	Concepts	Visual Features	Audio Features	Haptic Features	Proprioceptive Features
Nouns	Box	++-+-+-+----+-------------------------*----+---------------++---	----------*	*-*--*	******
	Ball	++------+++-------------------------+--++-----------------++++	*----------	--+---	******
	Cylinder	+++--------*+-------------------------++++----------------++---	*******--****	+*****	-+++--
	Cup	++-+-+-----++-------------------------++++----------------+----	*-*-*---*-***	+-----	-+++--
	Plate	******--+-+--+*-----------**------*--**--+-*----------------+++--	----------++	+-----	-***--
	Tool	*****+---+---+-------------------------+++----------------++---	++++++++++++	+-----	-+++--
Adjectives	Hard	******--+---++-------------------------++++----------------++++-	************	+***-+	-+++--
	Soft	++-+-+-++-+++-------------------------++++++-+----------------+****	+-+-+-+-+-+-+	+-+--*	++++++
	Noisy	**+***-+*****+--------------++----------+++++++----------------*****	++++*+++*+****	+-+--+	******
	Silent	******-**+--+*-------------------------+++*+++----------------*****	----+---++--++	*****	******
	Short	++---+-**--**-------------------------*-+**----------------++***	***-*---*-***	+-*--*	*****
	Tall	******-----***-------------------------+++----------------++---	************	+*****	-+++--
	Thick	+++**+-**--**-------------------------*-++----------------*****	*********-***	+-*-**	******
	Thin	+++--------*+-------------------------++++----------------++---	**--***-*****	+***--	-+++--
	Edgy	++-+-+-++--++-------------------------*-++---+----------------+++--	------+-+-+-*	+-*-**	******
	Round	+++++++--*---+*-------------------------++++----------------*****	************	+***-+	******
Verbs	Grasp	00+-0-*--000+00----------0-----000-00--0----0--------------00+++	None		
	Knock Down	0+0000+0-000000----------0-----000-00--0----0--------------+++++	None		
	Move Left	0-0000+0-000+00----------0-----000-00--0-00-0--------------+++++	None		
	Move Right	0+0000+--00+++0----------0-----000-00--000--0--------------+++++	None		
	Move Forward	-00000+--000++0----------0-----000-00--000000--------------+++++	None		
	Move Backward	+00000+0-00++00----------0-----000-00--000--0--------------+++++	None		
	Push Left	000-0-*0-000+00----------0-----000-00-------0--------------+++++	None		
	Push Right	0+0000+--00+++0----------0-----000-00--000-00--------------+++++	None		
	Push Forward	-00000*--000+00----------0-----000-00--0----0--------------+++++	None		
	Push Backward	+00-0-*0-000+00----------0-----000-00--00---0--------------0++++	None		
	Drop	***-00*0-000000----------0----0000-00--0----0--------------00+++	None		
	Throw	*0*000*0-000+00----------0-----000-00--0----0--------------0++++	None		
	Shake	000000*0-000000----------0-----000000--0-00-0--------------0++++	None		

References

[1] Ackerman, C. and Itti, L. (2005). Robot steering with spectral image information. *IEEE Trans. on Robotics*, 21(2):247–251.

[2] Akman, V. and Surav, M. (1996). Steps toward formalizing context. *AI magazine*, 17(3):55.

[3] Anand, A., Koppula, H. S., Joachims, T., and Saxena, A. (2012). Contextually guided semantic labeling and search for three-dimensional point clouds. *The International Journal of Robotics Research*, page 0278364912461538.

[4] Antoniak, C. E. (1974). Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The annals of statistics*, pages 1152–1174.

[5] Araki, T., Nakamura, T., Nagai, T., Nagasaka, S., Taniguchi, T., and Iwahashi, N. (2012). Online learning of concepts and words using multimodal lda and hierarchical pitman-yor language model. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, pages 1623–1630.

[6] Bando, T., Takenaka, K., Nagasaka, S., and Taniguchi, T. (2013). Automatic drive annotation via multimodal latent topic model. In *IROS*, pages 2744–2749.

[7] Barsalou, L. W. (2009). Simulation, situated conceptualization, and prediction. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1521):1281–1289.

[8] Barwise, J. and Perry, J. (1983). *Situation and Attitudes*. MIT Press.

[9] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022.

[10] Borghi, A., Flumini, A., Cimatti, F., Marocco, D., and Scorolli, C. (2011). Manipulating objects and telling words: a study on concrete and abstract words acquisition. *Frontiers in psychology*, 2.

[11] Brooks, R. A. (1991). Intelligence without representation. *Artificial intelligence*, 47(1):139–159.

[12] Butler, J. and Rovee-Collier, C. (1989). Contextual gating of memory retrieval. *Developmental Psychobiology*, 22(6):533–552.

[13] Bylander, T. (1991). Complexity results for planning. In *IJCAI*, volume 10, pages 274–279.

[14] Canini, K. R., Shi, L., and Griffiths, T. L. (2009). Online inference of topics with latent dirichlet allocation. In *International conference on artificial intelligence and statistics*, pages 65–72.

[15] Celikkanat, H., Orhan, G., and Kalkan, S. (2014a). A probabilistic concept web on a humanoid robot. IEEE Transactions on Autonomous Mental Development (under revision), available for reviewers at: http://www.kovan.ceng.metu.edu.tr/{%7E}sinan/ConceptWeb-ArticleUnderRevision.pdf.

[16] Celikkanat, H., Orhan, G., Pugeault, N., Guerin, F., Sahin, E., and Kalkan, S. (2014b). Learning and using context on a humanoid robot using latent dirichlet allocation. In *IEEE International Conference on Development and Learning and on Epigenetic Robotics (ICDL-EpiRob 2014)*.

[17] Chao, L. L. and Martin, A. (2000). Representation of manipulable man-made objects in the dorsal stream. *Neuroimage*, 12(4):478–484.

[18] Chapman, D. (1987). Planning for conjunctive goals. *Artificial intelligence*, 32(3):333–377.

[19] Choi, M. J., Lim, J. J., Torralba, A., and Willsky, A. S. (2010). Exploiting hierarchical context on a large database of object categories. In *CVPR*, pages 129–136. IEEE.

[20] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

[21] Coventry, K. R., Cangelosi, A., Newstead, S. N., and Bugmann, D. (2010). Talking about quantities in space: Vague quantifiers, context and similarity. *Language and Cognition*, 2(2):221–241.

[22] Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27.

[23] Cowan, N. (2004). *Working memory capacity*. Psychology Press.

[24] Creem, S. H. and Proffitt, D. R. (2001). Grasping objects by their handles: a necessary interaction between cognition and action. *Journal of Experimental Psychology: Human Perception and Performance*, 27(1):218.

[25] Damasio, A. R. (1989). Time-locked multiregional retroactivation: A systems-level proposal for the neural substrates of recall and recognition. *Cognition*, 33(1):25–62.

[26] Deacon, T. (1997). The symbolic species: the co-evolution of language and the human brain.

[27] Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407.

[28] Escobar, M. D. and West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the american statistical association*, 90(430):577–588.

[29] Friedman, S. L. and Scholnick, E. K. (2014). *The developmental psychology of planning: Why, how, and when do we plan?* Psychology Press.

[30] Gabora, L., Rosch, E., and Aerts, D. (2008). Toward an ecological theory of concepts. *Ecological Psychology*, 20(1):84–116.

[31] Gabrieli, J. D., Poldrack, R. A., and Desmond, J. E. (1998). The role of left prefrontal cortex in language and memory. *Proceedings of the national Academy of Sciences*, 95(3):906–913.

[32] Gärdenfors, P. (2004). *Conceptual spaces: The geometry of thought*. MIT press.

[33] Girolami, M. and Kabán, A. (2005). Sequential activity profiling: latent dirichlet allocation of markov chains. *Data Mining and Knowledge Discovery*, 10(3):175–196.

[34] Goldberg, R. F., Perfetti, C. A., and Schneider, W. (2006). Perceptual knowledge retrieval activates sensory brain regions. *The Journal of Neuroscience*, 26(18):4917–4921.

[35] Gouws, A. (2010). *A Python implementation of graphical models*. PhD thesis, Stellenbosch: University of Stellenbosch.

[36] Gregoriades, A., Obadan, S., Michail, H., Papadopoulou, V., and Michael, D. (2010). A robotic system for home security enhancement. *Aging Friendly Technology for Health and Independence*, pages 43–52.

[37] Griffiths, T. and Ghahramani, Z. (2005). Infinite latent feature models and the indian buffet process. In *Advances in Neural Information Processing Systems (NIPS)*.

[38] Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235.

[39] Hahn, U. and Chater, N. (1997). Concepts and similarity. *Knowledge, concepts and categories*, pages 43–92.

[40] Hampton, J. A. (1997). Conceptual combination. *Knowledge, concepts, and categories*, pages 133–159.

[41] Hastorf, A. H. and Cantril, H. (1954). They saw a game; a case study. *The Journal of Abnormal and Social Psychology*, 49(1):129.

[42] Hendler, J. A., Tate, A., and Drummond, M. (1990). Ai planning: Systems and techniques. *AI magazine*, 11(2):61.

[43] Heskes, T. et al. (2003). Stable fixed points of loopy belief propagation are minima of the bethe free energy. *Advances in neural information processing systems*, 15:359–366.

[44] Hoffman, M., Bach, F. R., and Blei, D. M. (2010). Online learning for latent dirichlet allocation. In *advances in neural information processing systems*, pages 856–864.

[45] Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57. ACM.

[46] Huth, A. G., Nishimoto, S., Vu, A. T., and Gallant, J. L. (2012). A continuous semantic space describes the representation of thousands of object and action categories across the human brain. *Neuron*, 76(6):1210–1224.

[47] Kalénine, S., Bonthoux, F., and Borghi, A. M. (2009). How action and context priming influence categorization: a developmental study. *British Journal of Developmental Psychology*, 27(3):717–730.

[48] Kalkan, S., Dag, N., Yürüten, O., Borghi, A. M., and Sahin, E. (2014). Verb concepts from affordances. *Interaction Studies*, 15(1):1–37.

[49] Kamp, H. and Reyle, U. (1993). *From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory*. Kluwer Academic (Dordrecht; Boston).

[50] Kellenbach, M. L., Brett, M., and Patterson, K. (2001). Large, colorful, or noisy? attribute-and modality-specific activations during retrieval of perceptual attribute knowledge. *Cognitive, Affective, & Behavioral Neuroscience*, 1(3):207–221.

[51] Kim, J. (1984). Concepts of supervenience. *Philosophy and Phenomenological Research*, pages 153–176.

[52] Kindermann, R., Snell, J. L., et al. (1980). *Markov random fields and their applications*, volume 1. American Mathematical Society Providence, RI.

[53] Klippel, A. and Montello, D. R. (2007). Linguistic and nonlinguistic turn direction concepts. In *Spatial information theory*, pages 354–372. Springer.

[54] Koenderink, J. J. and van Doorn, A. J. (1992). Surface shape and curvature scales. *Image and vision computing*, 10(8):557–564.

[55] Kruschke, J. K. and Johansen, M. K. (1999). A model of probabilistic category learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 25(5):1083.

[56] Lambon Ralph, M. A. (2014). Neurocognitive insights on conceptual knowledge and its breakdown. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1634):20120392.

[57] Lambon Ralph, M. A., Sage, K., Jones, R. W., and Mayberry, E. J. (2010). Coherent concepts are computed in the anterior temporal lobes. *Proceedings of the National Academy of Sciences*, 107(6):2717–2722.

[58] Larochelle, B., Kruijff, G.-J., Smets, N., Mioch, T., and Groenewegen, P. (2011). Establishing human situation awareness using a multi-modal operator control unit in an urban search & rescue human-robot team. In *Int. Symp. on Robot and Human Interactive Comm.*, pages 229–234.

[59] Lindemann, O., Stenneken, P., Van Schie, H. T., and Bekkering, H. (2006). Semantic activation in action planning. *Journal of Experimental Psychology: Human Perception and Performance*, 32(3):633.

[60] Liu, B., Liu, L., Tsykin, A., Goodall, G. J., Green, J. E., Zhu, M., Kim, C. H., and Li, J. (2010). Identifying functional mirna–mrna regulatory modules with correspondence latent dirichlet allocation. *Bioinformatics*, 26(24):3105–3111.

[61] Marszalek, M., Laptev, I., and Schmid, C. (2009). Actions in context. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pages 2929–2936. IEEE.

[62] Martin, A. (2007). The representation of object concepts in the brain. *Annu. Rev. Psychol.*, 58:25–45.

[63] McCarthy, J. (1989). Artificial intelligence, logic and formalizing common sense. *Philosophical logic and artificial intelligence*.

[64] McCarthy, J. (2007). From here to human-level ai. *Artificial Intelligence*, 171(18):1174 – 1182.

[65] Metta, G., Sandini, G., Vernon, D., Natale, L., and Nori, F. (2008). The icub humanoid robot: an open platform for research in embodied cognition. In *Proceedings of the 8th workshop on performance metrics for intelligent systems*, pages 50–56. ACM.

[66] Misra, D. K., Sung, J., Lee, K., and Saxena, A. (2014). Tell me Dave: Context-sensitive grounding of natural language to manipulation instructions. In *Robotics Science and Systems (RSS 2014)*.

[67] Mitchell, T. M., Shinkareva, S. V., Carlson, A., Chang, K.-M., Malave, V. L., Mason, R. A., and Just, M. A. (2008). Predicting human brain activity associated with the meanings of nouns. *Science*, 320(5880):1191–1195.

[68] Mummery, C. J., Patterson, K., Price, C., Ashburner, J., Frackowiak, R., Hodges, J. R., et al. (2000). A voxel-based morphometry study of semantic dementia: relationship between temporal lobe atrophy and semantic memory. *Annals of neurology*, 47(1):36–45.

[69] Nakamura, T., Araki, T., Nagai, T., and Iwahashi, N. (2011). Grounding of word meanings in latent dirichlet allocation-based multimodal concepts. *Advanced Robotics*, 25(17):2189–2206.

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

[71] Natraj, N., Poole, V., Mizelle, J., Flumini, A., Borghi, A. M., and Wheaton, L. A. (2012). Context and hand posture modulate the neural dynamics of tool–object perception. *Neuropsychologia*, 51:506–519.

[72] Nyga, D., Balint-Benczedi, F., and Beetz, M. (2014). PR2 looking at things: Ensemble learning for unstructured information processing with markov logic networks. In *ICRA*.

[73] Oden, G. C. (1987). Concept, knowledge, and thought. *Annual Review of Psychology*, 38(1):203–227.

[74] Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175.

[75] Orhan, G., Olgunsoylu, S., Sahin, E., and Kalkan, S. (2013). Co-learning nouns and adjectives. In *IEEE Third Joint International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob 2013)*, pages 1–6.

[76] Patterson, K., Nestor, P. J., and Rogers, T. T. (2007). Where do you know what you know? the representation of semantic knowledge in the human brain. *Nature Reviews Neuroscience*, 8(12):976–987.

[77] Pinto, N., Cox, D. D., and DiCarlo, J. J. (2008). Why is real-world visual object recognition hard? *PLoS computational biology*, 4.

[78] Pugeault, N. and Bowden, R. (2010). Learning pre-attentive driving behaviour from holistic visual features. *ECCV*, pages 154–167.

[79] Pulvermüller, F. (2002). *The neuroscience of language: on brain circuits of words and serial order*. Cambridge University Press.

[80] Pulvermüller, F. (2005). Brain mechanisms linking language and action. *Nature Reviews Neuroscience*, 6(7):576–582.

[81] Qin, A. K. and Suganthan, P. N. (2004). Robust growing neural gas algorithm with application in cluster analysis. *Neural Networks*, 17(8-9):1135–1148.

[82] Renninger, L. W. and Malik, J. (2004). When is scene identification just texture recognition? *Vision research*, 44(19):2301–2311.

[83] Robson, H., Zahn, R., Keidel, J. L., Binney, R. J., Sage, K., and Ralph, M. A. L. (2014). The anterior temporal lobes support residual comprehension in wernicke's aphasia. *Brain*, 137(3):931–943.

[84] Rosch, E. H. (1973). Natural categories. *Cognitive psychology*, 4(3):328–350.

[85] Rosseel, Y. (2002). Mixture models of categorization. *Journal of Mathematical Psychology*, 46(2):178–210.

[86] Roy, A. (2014). On findings of category and other concept cells in the brain: Some theoretical perspectives on mental representation. *Cognitive Computation*, pages 1–6.

[87] Salton, G. and McGill, M. J. (1983). *Introduction to modern information retrieval*. McGraw-Hill.

[88] Schank, R. C. and Abelson, R. P. (1977). Scripts, plans, goals and understanding. *Hillsdale, NJ: Lawrence Erlbaum*.

[89] Simmons, W. and Martin, A. (2009). The anterior temporal lobes and the functional architecture of semantic memory. *Journal of the International Neuropsychological Society*, 15(05):645–649.

[90] Steels, L. (2007). The recruitment theory of language origins. In Nehaniv, C. L., Lyon, C., and Cangelosi, A., editors, *Emergence of communication and language*, pages 129–150. Springer.

[91] Stein, N. L. and Trabasso, T. (1981). What's in a story: An approach to comprehension and instruction. *R. Glaser (Ed.), Advances in instructional psychology*, pages 213–267.

[92] Stoytchev, A. (2009). Some basic principles of developmental robotics. *Autonomous Mental Development, IEEE Transactions on*, 1(2):122–130.

[93] Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., and Rother, C. (2006). A comparative study of energy minimization methods for markov random fields. In *Computer Vision–ECCV 2006*, pages 16–29. Springer.

[94] Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006a). Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476).

[95] Teh, Y. W., Newman, D., and Welling, M. (2006b). A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In *Advances in neural information processing systems*, pages 1353–1360.

[96] Timpf, S., Volta, G. S., Pollock, D. W., and Egenhofer, M. J. (1992). A conceptual model of wayfinding using multiple levels of abstraction. In *Theories and methods of spatio-temporal reasoning in geographic space*, pages 348–367. Springer.

[97] Torralba, A. (2003). Contextual priming for object detection. *International Journal of Computer Vision*, 53(2):169–191.

[98] Tsuruma, G., Kanai, H., Nakada, T., and Kunifuji, S. (2007). Dangerous situation awareness support system for elderly people with dementia. In *Int. Conf. on Human Computer Interaction*, pages 62–67. ACTA Press.

[99] Tucker, M. and Ellis, R. (2001). The potentiation of grasp types during visual object categorization. *Visual cognition*, 8(6):769–800.

[100] Van Deemter, K., Krenn, B., Piwek, P., Klesen, M., Schröder, M., and Baumann, S. (2008). Fully generated scripted dialogue for embodied agents. *Artificial Intelligence*, 172(10):1219–1244.

[101] van Elk, M., van Schie, H., and Bekkering, H. (2013). Action semantics: a unifying conceptual framework for the selective use of multimodal and modality-specific object knowledge. *Physics of life reviews*.

[102] Veldhuizen, T. (2007). Ubigraph: Free dynamic graph visualization software.

[103] Wang, C., Paisley, J. W., and Blei, D. M. (2011). Online variational inference for the hierarchical dirichlet process. In *International Conference on Artificial Intelligence and Statistics*, pages 752–760.

[104] Xing, D. and Girolami, M. (2007). Employing latent dirichlet allocation for fraud detection in telecommunications. *Pattern Recognition Letters*, 28(13):1727–1734.

[105] Yeh, W. and Barsalou, L. W. (2006). The situated nature of concepts. *The American journal of psychology*, pages 349–384.

[106] Yoon, E. Y., Humphreys, G. W., and Riddoch, M. J. (2010). The paired-object affordance effect. *Journal of Experimental Psychology: Human Perception and Performance*, 36(4):812.

[107] Zhai, K. and Boyd-Graber, J. (2013). Online latent dirichlet allocation with infinite vocabulary. In *Proceedings of The 30th International Conference on Machine Learning*, pages 561–569.