

BUILDING A WEB OF CONCEPTS ON A HUMANOID ROBOT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÜNER ORHAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

AUGUST 2014

Approval of the thesis:

BUILDING A WEB OF CONCEPTS ON A HUMANOID ROBOT

submitted by **GÜNER ORHAN** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Adnan Yazıcı
Head of Department, **Computer Engineering**

Assist. Prof. Dr. Sinan Kalkan
Supervisor, **Computer Engineering Department, METU**

Examining Committee Members:

Prof. Dr. Göktürk Üçoluk
Computer Engineering Department, METU

Assist. Prof. Dr. Sinan Kalkan
Computer Engineering Department, METU

Assoc. Prof. Dr. Erol Şahin
Computer Engineering Department, METU

Assist. Prof. Dr. Uluç Saranlı
Computer Engineering Department, METU

Assist. Prof. Dr. Sanem Sariel-Talay
Computer Engineering Department, İTÜ

Date:

13.08.2014

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: GÜNER ORHAN

Signature :

ABSTRACT

BUILDING A WEB OF CONCEPTS ON A HUMANOID ROBOT

Orhan, Güner

M.S., Department of Computer Engineering

Supervisor : Assist. Prof. Dr. Sinan Kalkan

August 2014, 76 pages

In this thesis, an effective approach for predicting nouns, adjectives and verbs is introduced for more effective communication between a humanoid robot and a human actor. There are three important challenges addressed by our approach: The first one is the accurate prediction of words in language. Most of the existing robotics studies predict words in language using perceptual information only. However, due to noise and ambiguity in low-level sensory information, prediction using perceptual information is often incorrect. The second challenge is the meaning of the words. The existing studies mostly use discriminative methods to predict words, yet the underlying semantics of what, e.g., a certain noun represents, is not adequately addressed in the literature. The third challenge is representation of the relations between the different words in language. It is known that humans activate in their brains not only the meaning of the word when that word is uttered but also the related words and their meaning. However, this challenge has not been addressed in the robotics literature. In this thesis, the words in language are first conceptualized and gradually, a web of concepts is built from the interactions of the robot. The web is built using the co-occurrence information of words, modeled as a Markov Random Field and trained using Loopy Belief Propagation, a widely-used method for such tasks. The thesis shows on iCub, a humanoid robot, that such a web of concepts addresses to a certain extent all the challenges discussed above: the web improves prediction of word categories; it represents the meaning of words in concepts, and it represents the

relations between the words and their meaning. As such, this thesis makes a first important step towards grounded representation of a semantic network on a humanoid robot, which can be used for several high-level cognitive tasks, such as contextual reasoning, planning, language understanding, etc.

Keywords: Cognitive Robotics, Conceptualization, Symbol-Grounding, Web of Concepts

ÖZ

İNSANSI ROBOTTA KAVRAM AĞI OLUŞUMU

Orhan, Güner

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi : Yrd. Doç. Dr. Sinan Kalkan

Ağustos 2014 , 76 sayfa

Bu tezde, insansı robot ve insan arasında daha etkili bir iletişim sağlamak için isim, sıfat ve fiilleri tahmin etmede kullanılacak bir yaklaşım önerilmiştir. Yaklaşımımızın hedef aldığı üç önemli nokta bulunmaktadır: Bunlardan ilki, dildeki kelimelerin doğru tahmin edilmesidir. Robotik çalışmaların çoğu, sadece algısal bilgiyi kullanarak dildeki kelimeleri tahmin etmektedir, ancak algısal özellikler kullanılarak yapılan tahminler, sensör bilgisindeki gürültü ve belirsizlikler nedeniyle doğru değildir. İkinci nokta ise kelimelerin anlamlarıdır. Var olan çalışmalarda, çoğunlukla kelimeleri tahmin etmek için ayrimsal yöntemler kullanılmıştır. Fakat, belli bir ismin neyi ifade ettiğinden, literatürde yeteri kadar bahsedilmemiştir. Üçüncü nokta ise, dildeki farklı kelimelerin arasındaki ilişkinin gösterilişidir. İnsan beyninde bir kelime söylendiği zaman sadece o kelimenin anlamı değil, aynı zamanda alakalı kelimeler ve anlamlarının da aktif olduğu bilinmektedir. Ancak, bu duruma robot literatüründe değinilmemiştir. Bu tezde, ilk olarak dildeki kelimeleri kavramsallaştırılıp, devamında robotun hareketlerinden kavram ağı oluşturulmaktadır. Bu ağ, kelimelerin eş-oluş bilgilerinin, Raslantısal Markov Alanı (Markov Random Field) ile modellenmesi ve bu tarzda çalışmalarda yaygın olarak kullanılan, Döngüsel Fikir Aktarımı (Loopy Belief Propagation) kullanılarak öğretilmesi ile geliştirilmiştir. Bu tez, kavram ağının yukarıda bahsedilen eksikliklerini belli bir ölçüde giderdiğini, iCub isimli insansı robot üzerinde göstermektedir. Ağ, kelime kategori tahminlerini geliştirmekte, kavram kelimelerinin anlamlarını göstermekte ve kelimelerin anlamları arasındaki bağlantıyı

ifade etmektedir. Tüm bunlar, içeriksel anlamlandırma, dili anlama gibi konuları içeren yüksek seviyeli bilişsel konularda kullanılabilecek olan insansı robottaki anlamsal ağın temellendirilmiş gösterimini sunan bu tezi önemli bir aşama yapmaktadır.

Anahtar Kelimeler: Bilişsel Robotik, Kavramsallaştırma, Sembol Temellendirme, Kavram Ağı

To my family and beloved people in my life

ACKNOWLEDGMENTS

First of all, I would like to thank to my family for supporting me and not making me feel lonely, especially my mom, dad, and precious grandfather. They are the best family in the world. I could not have completed my M.Sc. education without their supports.

I would like to express my respect and thank to my advisor Sinan Kalkan for always supporting and guiding me at the whole study. He is very patient and responsible, and instil determination and confidence in me.

I'd like to thank Erol Şahin, the head of our laboratory and one of the best lecturers in our computer engineering department. He is very instructive. Moreover, I'd like to show my gratitude to him for invaluable efforts for founding the KOVAN laboratory and bringing iCub for research.

My special thanks go to my lab friends and colleagues who made laboratory a better place to live and work. I'd like to thank Sertaç Olgunsoylu for his friendship and invaluable advices. He is a very clever and humorous person I've ever met. I am really lucky to know Yiğit Çalışkan because he is the one of my best friends. I always feel his support. I really feel special to meet Fariba Yousefi. I would like to thank her for permanent smiling face and introducing special one to me, Mustafa Parlaktuna for his deep knowledge in iCub and attitude that makes you feel comfortable in case of unexpected problems, Kadir Fırat Uyanık for his excellence in robotics and software libraries, Asil Kaan Bozcuoğlu for his companionship and different sense of humor, and Hande Çelikkanat for her valuable advices and great knowledge in cognitive science. She is also so helpful. Moreover, I thank Osman Tursun for his ridiculous Turkish, Mehmet Akif Akkuş for his interesting knowledge in human health, and Fatih Gökçe for tolerating my jokes.

Moreover, my gratitude goes to my friends outside the lab. Initially, I'd like to thank İpek Sırım for supporting me and always making me happy and refreshed in last one year. I cannot express my feelings for her with bare words. She is so special for me. I'd like to thank my perfect friend Talha Koruk. We will be friends forever. I'd like to thank my old and precious friend Özgün Karaahmetoğlu. I can feel his support even if he is not in my town.

This work is partially funded by the Turkish National Science and Technology Organization (TÜBİTAK) under the project no 111E287.

TABLE OF CONTENTS

ABSTRACT	v
ÖZ	vii
ACKNOWLEDGMENTS	x
TABLE OF CONTENTS	xi
LIST OF TABLES	xv
LIST OF FIGURES	xvii
LIST OF ALGORITHMS	xx
LIST OF ABBREVIATIONS	xxi
CHAPTERS	
1 INTRODUCTION	1
1.1 Contributions	2
1.2 Organization	4
2 BACKGROUND AND LITERATURE SURVEY	5
2.1 Language Studies in Robots	5
2.2 The Notion and Theories of Conceptualization	7
2.3 Hierarchy vs. Web of Concepts	8

2.3.1	Hierarchy of Concepts	8
2.3.2	Web of Concepts	11
2.4	Concept Studies in Robots	12
2.5	Probabilistic Graphical Models	15
2.5.1	Elimination Algorithm and Belief Propagation . .	17
2.5.2	Loopy Belief Propagation	19
2.6	Support Vector Machine (SVM)	22
3	EXPERIMENTAL SETUP AND CONCEPTUALIZATION	23
3.1	Experimental Setup	23
3.1.1	Hardware Components	23
3.1.1.1	iCub Humanoid Robot Platform . . .	23
3.1.1.2	Kinect	24
3.1.2	Software Components	25
3.1.2.1	iCub Modules	25
3.1.2.2	Yet Another Robot Platform (YARP) .	25
3.1.2.3	Point Cloud Library (PCL)	25
3.1.2.4	Ubigraph	26
3.2	Perception	26
3.2.1	Features	26
3.3	Data Collection	30
3.4	Feature Extraction	32

3.5	Conceptualization of the Categories	32
3.6	Category prediction	33
3.6.1	Prediction of Noun Categories	34
3.6.2	Prediction of Adjective Categories	34
3.6.3	Prediction of Verb Categories	34
3.6.4	Prediction of Effects Using SVM	35
4	EFFECT OF CO-OCCURRENCE ON CATEGORY PREDICTION .	37
4.1	Contribution of co-occurrence information	37
4.2	Cross-Situational Labeling of Categories	38
4.3	Results	39
4.3.1	Noun and Adjective Prediction using SVM	40
4.3.2	Co-occurrence Effect on Prediction	40
4.3.3	The “What object is it?” Game	44
4.3.4	Concept Labeling Accuracy	45
5	A WEB OF CONCEPTS	47
5.1	Building a Web of Concepts	47
5.1.1	Integrating LBP into Web of Concepts	48
5.2	Results	50
5.2.1	Scenario 1: “Perception-Driven Activation of Concepts in the Web”	50
5.2.2	Scenario 2: “Interaction-Driven Activation of Concepts in the Web”	54

5.2.3	Scenario 3: “Command-Driven Activation of Concepts in the Web”	58
6	DISCUSSION AND CONCLUSION	59
6.1	Limitations and Future Work	60
	REFERENCES	63
	APPENDICES	
A	HIERARCHICAL CONCEPT FORMATION METHODS	71
B	MODIFIED CROSS-SITUATIONAL LABELING	75
B.1	Background	75
B.2	Algorithm	76

LIST OF TABLES

TABLES

Table 3.1	Features extracted from the interactions with the environment. Parenthesized numbers indicate the index of features in the feature vector. . . .	27
Table 3.2	The possible effect outcomes of behaviors on different noun categories. Empty cells imply that these behaviors are not applied to the objects in the category. (<i>arg: Left, Right, Forward, Backward</i>)	30
Table 4.1	Prototypes of noun and adjective co-occurrences. ‘*’, ‘+’ and ‘-’ respectively represent inconsistent co-occurrence, consistent co-occurrence and consistent non-co-occurrences.	38
Table 4.2	Average noun and adjective prediction accuracy results on the training set.	40
Table 4.3	Predicted adjectives for some objects from the test set (bold denotes correct classification). The co-occurrence weight w_{na} is taken as 0.2 where prediction performance is maximized.	42
Table 4.4	Predicted nouns for some objects from the test set (bold denotes correct classification). The co-occurrence weight w_{an} is taken as 0.2. . . .	43
Table 4.5	“What object is it?” game: Determine noun based on given adjectives.	44
Table 5.1	Possible applicable set of behaviors with respect to object categories. (<i>arg: Left, Right, Forward, Backward; A:Applicable; NA: Not-Applicable</i>)	50

Table 5.2	The prediction accuracies of noun and adjective categories using the concept web, with respect to the perception-only guesses. 6 objects, one of each noun category, are used for demonstration. Images depict RGB-colored depth images (collected via PCL library from the Kinect sensor). The second and third columns depict the perception-only results, while the fourth and fifth columns show the concept web predictions. Prediction confidences are indicated in paratheses. The use of bold text indicates correct decisions. Striked-out text indicates wrong decisions. [Best viewed in color]	53
Table 5.3	The predictions of applicable behaviors and their likely effects via the concept web. <i>NA</i> stands for <i>Not-Applicable</i> . The confidence values of the predictions are unanimously (100%), so are intentionally not shown for clarity. Images depict RGB-colored depth images (collected via PCL library from the Kinect sensor). [Best viewed in color]	54
Table 5.4	The prediction accuracies of noun and adjective categories on the <i>novel</i> objects using the concept web. Initially, only visually activated concepts are perceptually predicted, and the activations are spread to predict all related concepts. 4 novel objects, are used for demonstration. The second and third columns depict the perception-only results, while the fourth and fifth columns show the concept web predictions. Prediction confidences are indicated in paratheses. The use of bold text indicates correct decisions. Striked-out text indicates wrong decisions. [Best viewed in color]	56
Table 5.5	The selection of objects on which sample commands are applicable behaviors. The confidence values of the predictions are unanimously (100%), so are intentionally not shown for clarity. Images depict RGB-colored depth images (collected via PCL library from the Kinect sensor). [Best viewed in color]	58
Table B.1	Initial tables for adjective and noun concepts labeling	75

LIST OF FIGURES

FIGURES

Figure 1.1 Existing approaches focus on learning methods for noun and adjective categories. Initially, we have used co-occurrence information for only noun and adjective categories. Finally, we have used the web of concepts structure in order to activate the most probable concept in the web. Verb categories are shown as green nodes in (c). [Best viewed in color] . . .	3
Figure 2.1 The exemplification of familiarization and discrimination operators in EPAM while incrementally creating tree. (Figure from [1])	9
Figure 2.2 Markov random field model with 2 maximal cliques and their potential tables. $\psi(\cdot)$ refers to the clique potential. [Best viewed in color] . . .	15
Figure 2.3 Sample Markov Random Field chain consisting of six variables . . .	17
Figure 2.4 Schema for message passing to find the probability $p(x_3)$ for Figure 2.3. (adapted from [2])	18
Figure 2.5 MRF graphs and its corresponding graph used in Loopy Belief Propagation. The circular nodes represents the maximal cliques while square nodes shows the variable nodes in initial MRF graph.	19
Figure 2.6 Separated cliques nodes with their separator nodes (sub-trees) of Figure 2.5	20
Figure 2.7 Sample initial four iteration for loopy belief propagation in Figure 2.5. Double star means that the value of potential table is updated twice. . .	21
Figure 2.8 Schema for dividing multi-dimensional feature space into two clusters with hyperplane using kernel functions of SVM. (Figure from [3]) . . .	22
Figure 3.1 iCub Humanoid Robot Platform with microphone and Kinect sensor (Figure taken from [4]).	24
Figure 3.2 iCub Humanoid Robot Platform coordinate reference frame.	24
Figure 3.3 The Kinect sensor	25

Figure 3.4	Content of effect and entity features	28
Figure 3.5	All objects which are separated according to their noun categories. [Best viewed in color]	29
Figure 3.6	Entity features extraction applying <i>grasp & shake</i> behavior in order of data collection procedure	31
Figure 4.1	Adjective prediction accuracy for the testing set with respect to weighted contribution of co-occurrence	41
Figure 4.2	Noun prediction accuracy for the testing set with respect to weighted contribution of co-occurrence	41
Figure 4.3	Accuracy result for concept labeling. The horizontal axis shows the percentage of included subsets of adjectives, while the vertical axis is the accuracy for correct labeling of concepts	45
Figure 5.1	MRF representation of our web of concepts (only perceptual and language concepts included) using Ubigraph Library [5]. For the sake of comprehensibility, the labels of the background concepts are not written explicitly.	49
Figure 5.2	Schematized representation of Scenario 1. The <i>Cup</i> is given to the system and all related concepts are activated. The action space and verb concepts are contoured with green, whereas blue and orange colors represent the noun and adjective categories for the object, respectively. The gray and smaller fonts show inactive concepts in the web, while big- ger fonts and colored nodes represent activated concepts. There are other concepts that are not shown for clarity. (<i>ML</i> : Move Left, <i>MR</i> : Move Right, <i>MF</i> : Move Forward, <i>MB</i> : Move Backward, <i>PL</i> : Push Left, <i>PR</i> : Push Right, <i>PF</i> : Push Forward, <i>PB</i> : Push Backward)	51
Figure 5.3	Schematized representation of Scenario 2. The <i>Ball</i> is given to the system and <i>Push</i> behavior is commanded to iCub. All related concepts are activated (only visually perceivable concepts). The action space and verb concepts are contoured with green, whereas blue and orange colors represent the noun and adjective categories for the object, respectively. The gray and smaller fonts show inactive concepts in the web, while big- ger fonts and colored nodes represent activated concepts. There are other concepts that are not shown for clarity. (<i>PL</i> : Push Left, <i>PR</i> : Push Right, <i>PF</i> : Push Forward, <i>PB</i> : Push Backward)	55

Figure 5.4 Schematized representation of Scenario 3. The sample <i>Ball</i> , <i>Cup</i> , and <i>Plate</i> objects are given to the system and <i>Drop</i> behavior is commanded to iCub. iCub selects any one of these objects if the commanded behavior is applicable. In this scenario, the <i>Ball</i> object is selected and its activated concepts are shown. The action space and verb concepts are contoured with green, whereas blue and orange colors represent the noun and adjective categories for the object, respectively. The gray and smaller fonts show inactive concepts in the web, while bigger fonts and colored nodes represent activated concepts. There are other concepts that are not shown for clarity.	57
Figure 6.1 Representation of Long and Short Term Memories consisting of “situated” concepts and instantaneous concepts related with behavior and object, respectively.	61
Figure A.1 COBWEB structure of each concept node. Each node is created in order of creation. Figure is taken from [1]	71
Figure A.2 The operators application procedures in COBWEB. Figure is taken from [6]	72

LIST OF ALGORITHMS

ALGORITHMS

Algorithm 1	Derivation of Prototypes. Algorithm from [7]	32
Algorithm 2	Modified Cross-Situational Based Labeling	76

LIST OF ABBREVIATIONS

RGNG	Robust Growing Neural Gas
PCL	Point Cloud Library
LDA	Latent Dirichlet Allocation
YARP	Yet Another Robot Platform
YAAFE	Yet Another Audio Feature Extractor
MFCC	Mel-Frequency Cepstral Coefficients
MRF	Markov Random Field
SVM	Support Vector Machine
LBP	Loopy Belief Propagation
PGM	Probabilistic Graphical Model

CHAPTER 1

INTRODUCTION

With advances in technology, robots will become an inevitable part of our daily lives. They are already used in many areas such as medical, search and rescue operations, industry, service sectors and even in our homes. Such challenging environments (will) require the robots to be adaptive, self-extensive and able to communicate properly with humans using daily spoken language. An important bottleneck for this goal is the acquisition and representation of information coming from the environment as well as the linking of such information to language.

There are many studies for teaching robots the words in language, e.g., nouns or adjectives, from sensorimotor data acquired from the interactions of the robot (e.g., [8, 9, 10]). In these studies, word categories are mostly learned, represented and predicted separately (Figure 1.1a). However, it is known that, in humans, when a word is heard, not only the meaning of that word but also the meanings of the related words are activated [11, 12]. For example, when you hear the word “apple”, the color “red” (an adjective) and “eat” (a verb) are also activated in our brains. Based on these, it has been suggested that the concepts corresponding to words in language are linked to each other in our brains, and there is actually a web of concepts, where related concepts activate and affect each other. A pioneering work is Mitchell et al.’s study [13], which demonstrates that fMRI activations for complex concepts (i.e, celery) can be predicted by accumulating known fMRI activations for related, simpler concepts (i.e, eat, taste, fill). Interestingly, the target fMRI activation for the complex concept is very close to a superposition of the simpler concept activations, weighted by their co-occurrences.

Motivated by such findings in Neuroscience and Psychology, this thesis studied the building, representation and use of a web of concepts on a humanoid robot. First, a set of nouns, adjectives and verbs in language are conceptualized by the robot first from the interactions with the objects in the environment. Then, the robot used the co-occurrence information between word categories to link them, to build a web of concepts that are linked to language, perceptual systems and the motor system of the robot. The web is modeled as a Markov Random Field (MRF) due to MRF's representational power and ease of making inferences, and trained using Loopy Belief Propagation. By propagating the beliefs about the concepts over the web, the activation of concepts in the web is achieved.

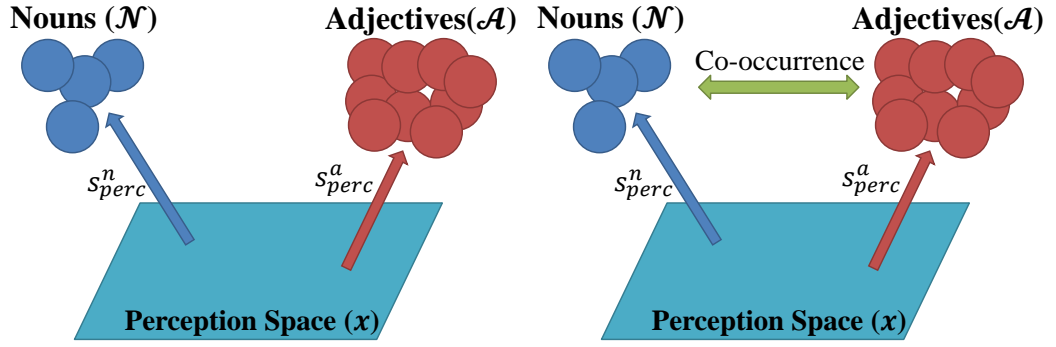
1.1 Contributions

The contributions of the thesis are summarized below:

- A. The first contribution of the thesis was to use the co-occurrence information between adjectives and nouns in language to improve their prediction accuracies (Figure 1.1b). It is known that predicting adjectives is harder than predicting nouns, and a robot makes many mistakes in adjective prediction [9]. However, the thesis shows that the co-occurrence information between nouns and adjectives can be used to improve the prediction of nouns and especially adjectives.
- B. The thesis extended the first contribution in Part A and modeled a web of concepts for a set of nouns, adjectives and verbs (Figure 1.1c) as a second contribution. The web was modeled using Markov Random Field and trained using Loopy Belief Propagation. The thesis shows that such a web improves the prediction of word categories in language, and with this web, the relevant concepts, words and actions can be activated in a similar way as in humans.

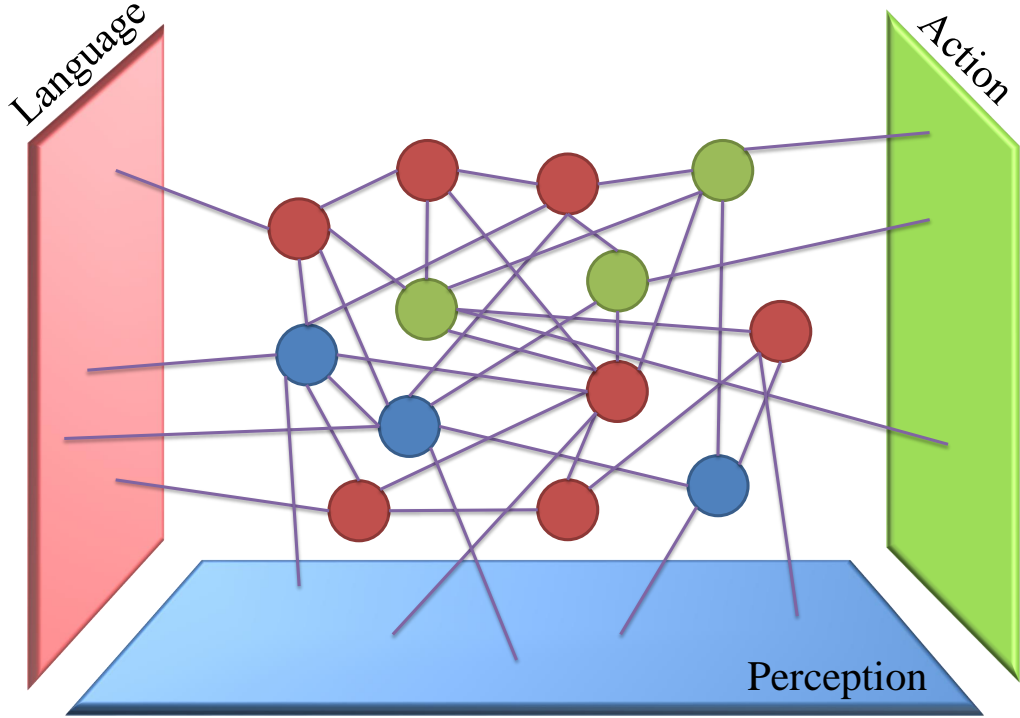
The work explained in this thesis are disseminated in the following:

- **Güner Orhan**, Hande Çelikkanat, and Sinan Kalkan, A Probabilistic Web of Concepts on a Humanoid Robot, *IEEE Transactions on Autonomous Mental Development* (Submitted)



(a) Existing work

(b) Contribution using co-occurrences



(c) Contribution using concept web

Figure 1.1: Existing approaches focus on learning methods for noun and adjective categories. Initially, we have used co-occurrence information for only noun and adjective categories. Finally, we have used the web of concepts structure in order to activate the most probable concept in the web. Verb categories are shown as green nodes in (c). [Best viewed in color]

- Hande Çelikkanat, **Güner Orhan**, Nicolas Pugeault, Frank Guerin, Erol Şahin and Sinan Kalkan, Learning and Using Context on a Humanoid Robot Using Latent Dirichlet Allocation, *IEEE Forth Joint International Conference on Development and Learning and on Epigenetic Robotics (ICDL - EpiRob)*, Genova, Italy, 2014 (Accepted)
- **Güner Orhan**, Sertaç Olgunsoylu, Erol Şahin and Sinan Kalkan, Co-learning Nouns and Adjectives, *IEEE Third Joint International Conference on Development and Learning and on Epigenetic Robotics (ICDL - EpiRob)*, Osaka, Japan, 2013

1.2 Organization

In chapter 2, the background about the methods used throughout this thesis is presented, and we give the current studies related with our contributions.

In chapter 3, we describe the experimental setup consisting of iCub Humanoid Robot platform and Kinect and give commonly used libraries to perform our experiments. Furthermore, we give the detailed information about the processes, such as feature extraction, conceptualization, and prediction of categories for a test object, in the following chapters.

In chapter 4, we will give description about the co-occurrence-based learning of adjective and nouns, and the results of this method will be demonstrated.

In chapter 5, we will mention about building a web of concepts using co-occurrence information, and provide scenarios to emphasize on the contribution by showing the results of them.

In chapter 6, the conclusion remarks will be given, and we will make some discussion how to enhance the current study. Finally, our intentions to overcome the deficiencies to make our system better.

CHAPTER 2

BACKGROUND AND LITERATURE SURVEY

This chapter reviews the language studies in robots, and discusses the different views of what concepts are. In addition, a discussion on the hierarchical and web-based representation of concepts is provided.

2.1 Language Studies in Robots

Language in cognitive robotics is an important keystone, providing the communication between a robot and a human. In the world of a robot, the learned concepts are nothing more than a set of representative information about categories. At this level, a robot is unaware of their semantic meanings, in other words, the referents of symbols (words) in human language.

In human development, infants start babbling at approximately six months old [14]. While growing up, they initially talk with pseudo-words, imitating the spoken words [15]. The actual language learning is a supervised process evolving throughout the entire life of a human. After learning how to speak and give symbols to the objects in the world, they start to comprehend the concepts interacting with them. The use of language shapes our knowledge of concepts and provides a better classification of objects. For example, Xu [16] made some experiments on 9-month infants, and showed them objects by giving the spoken labels. This work shows that two distinct labels for two distinct objects elicit the successful classification of objects. Furthermore, if a single label is given for different concepts, then the infants lump them into one category [17].

Learning and understanding the language require linking words with sensorimotor data. The most strongest support of this idea comes from the work of Harnad [18]. According to his theory, there is a gap between the words and their semantic meanings, and this cannot be fixed in a hard-coded manner. It is like learning foreign language from dictionary. Therefore, one should ground the symbols (words) to the sensor and motor interactions of the agent. The widely-accepted answer to this problem is the embodiment of the agent.

Following the above idea, roboticists have focused on teaching language to robots. For instance, Cangelosi [10] has made a study about grounding of language in cognitive agent and developmental robotics, by applying three different models: (i) a multi-agent modeling of language evolution, focusing on the interactions between agents in the same environment to find the edible foods (mushrooms), (ii) the model of transferring the symbol groundings between two robotics agents which are teacher and learner in three steps, namely basic action learning, which direct imitation of behavior with respect to a given object, entry-level naming, where the learner imitates action with respect to the linguistic, and higher-order learning, including the acquisition of the complex actions without using teacher, (iii) the comprehension of language using humanoid robot iCub, which is a process of relating the speech signals with behaviors and noun categories using neural network. Moreover, Cangelosi and Parisi [19] made an experiment in order to link nouns to vertical and horizontal bars, and also verbs to *pull* and *push* behaviors. They also realized that nouns cause more neural activity on hidden layer of feed-forward neural network, whereas verbs result in an activation around the synapses. In the famous experiment of Steel et al. [20, 21] named “Talking Heads”, the aim is to generate a shared lexicon and ground the words into perceptually gathered concepts in an unsupervised manner. There are two types of group which are “speakers” and “hearers”. Speakers firstly conceptualize the context by using the properties which separate it from the surroundings. Secondly, they have to say a word using their own lexicon of form-meaning (word-visual categories) success tables. Hearers should predict the correct context using the associations between meanings and words.

The language can also be used for cross-situational labeling of the concepts [22, 23, 24]. Concepts that are learned from experiences has no label unless they are linked to

words in language. This link provides an effective social interaction with surrounding organisms for a robot. To make a real world example, some other information is given to a robot to create reference of words with language using cross-situational correlations. For instance, the experiment conducted by Smith and Yu [22, 23] shows that the cross-situational learning approach for linking correlations between words and the referents is useful for learning the meaning of words even if these words have no meaning in human language and generated by a computer.

2.2 The Notion and Theories of Conceptualization

As a term, *concept* can be defined as the properties that represent a category. Another definition of the concept is the thought or similarity that allows us to classify current and previous situations [25]. They can be concrete like *orange* or abstract like *beauty*. There are different views for how concepts are learned or represented. The main well-accepted views are:

- **The Rule-based view** [26]: In this view, the categories are considered as a collection of all possible members with strict boundaries, and each novel object is a member of a category or not, meaning that there are strict boundaries to be selected as a member of any category. If properties of an object overlap with the common properties of a category, then we conclude that this object can be classified as a member of that category. The same description can also be found in the work of Medin et al. [27]. They emphasize that this view lacks the judgment of category membership of a new object due to strict boundaries in category properties. Therefore, for some cases, it is not possible to determine a category of an object due to its unclear properties.
- **The prototype-based view** [28]: In this view, prototypes show the best representative property of categories, and there are no strict rules to test whether an object belongs to a category or not. The postulated natural prototypes, representing a category best, can be learned with less errors than prototypes which are created using the outliers of any category. For the sake of clarification, the natural prototypes can be thought as the main property for a specific category,

and the features that constitute a concept are represented using the statistical distribution of this feature over entire members.

- **The Exemplar-based View** [29]. In this view, a category is represented by its examples not the properties. Novel objects can be added to the list of members of a category if they are similar to one of the members of that category. For instance, the members of a *BALL* concept might be:

$$BALL = \left\{ \text{red ball} \quad \text{purple ball} \quad \text{orange ball} \quad \text{yellow ball} \quad \text{green ball} \quad \dots \right\}$$

A new object is a *BALL* if it is similar to one of the *BALL* exemplars. However, this view cannot determine which properties of a concept best describes it [27].

Although these views can be seen as the basis for the conceptualization theories, There is another approach that treats a concept as a hybrid representation [30]. With the guidance of these theories, conceptualization is still an active research area, that must be carefully investigated [31] since we cannot perform any action without concepts [32].

2.3 Hierarchy vs. Web of Concepts

Although concepts can be learned independently from each other, they are linked to each other in our brains. Therefore, there is the issue of how links between concepts can be represented.

2.3.1 Hierarchy of Concepts

As mentioned in the work of Gennari and his colleagues [1], human learning can be thought as a process of concept formation. Humans learn new concepts by setting a concept hierarchy while observing or experiencing new objects, behaviors or events. The place of the concepts in hierarchy can be determined with respect to generality of them. The more general concepts reside in the upper parts of the hierarchy, while more specific concepts are located closer to the leaf nodes. The main aim in this

concept formation process as a hierarchical taxonomy is to understand the world and make predictions.

The classification in hierarchy commences at root node, and goes deeper to the branches, which can be one or multiple selection of branches. Another important issue to be determined is the creation of these branches. The number of branching can be specified by a teacher (supervised learning) or the algorithm decides it with respect to different criteria (unsupervised learning). Although the information of classes is given by an actor, Quinlan's decision tree approach [33] can be considered as unsupervised learning of branches, since, the system determines the sub-classes of any class in a tree. Another important feature is the incremental Hill-climbing method [1]. Hill-climbing is a well-known search method. After determining the current state of search, the algorithm relates an instance with the sub-states (possible candidates for iteration) using some evaluation function. The most correlated sub-state is determined, and the same procedure is applied recursively for all sub-trees until there is no possible move.

After mentioning some properties of hierarchical concept formation, we would like to present some well-known hierarchical concept formation studies in this area.

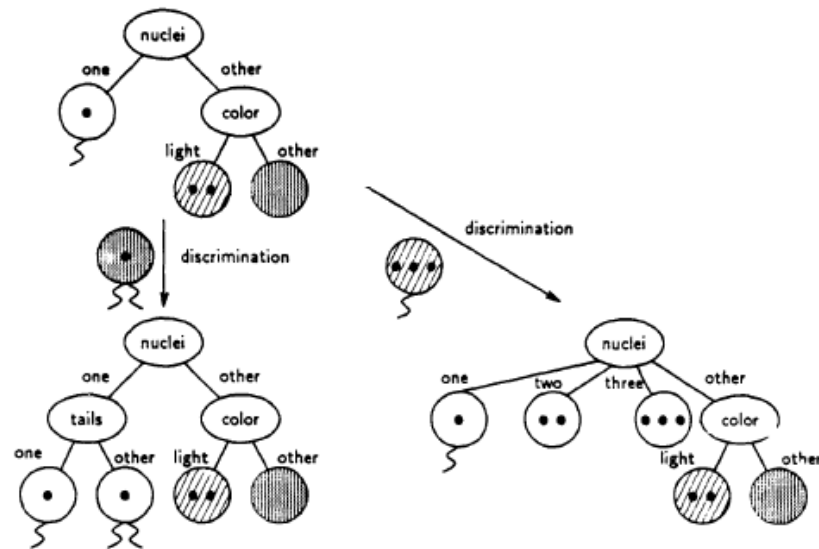


Figure 2.1: The exemplification of familiarization and discrimination operators in EPAM while incrementally creating tree. (Figure from [1])

Elementary Perceiver And Memorizer (EPAM) [34, 35]: This method is almost the oldest incremental concept formation method. Each node in the tree is a combi-

nation of attribute-value pairs. The internal nodes include testing criteria, which must be evaluated for an instance to go down in the hierarchy. There are two possible operations namely *familiarization* and *discrimination*. The familiarization is a process of adding an attribute-value pair to the node due to more specific properties of a new instance, which is also common for the instances belonging to that node, while the discrimination is a process of splitting the node and placing the instances with respect to a newly defined test procedure or property value pair on the split node (Figure 2.1). The selection of the attribute for testing is a random process. Although EPAM has some powerful features such as the leading in incremental concept formation methods in machine learning, it has some drawbacks. The representative images of the instances are only at terminal nodes constraining with concept hierarchy. Moreover, concepts have strict boundaries but according to the view of prototype-based concepts [28], there are main properties (prototypes) representing the concepts, and these do not have strict boundaries.

UNiversal MEMory (UNIMEM) [36]: As was the case in the above method, each node consists of attribute-value pairs. However, there are two more numeric values for each pair. These are feature frequency and the confidence value for each feature. The former one demonstrates the frequency of this feature in other generalizations, indicating the relevance of the generalization for new instances, and the latter shows the confidence of a feature for a generalization. Moreover, concepts can be placed not only in terminal nodes but also in internal nodes. Although UNIMEM can be utilized to simulate the human learning, it is basically developed for clustering the large chunks of data in the memory and retrieve relevant information or data by user queries. Unlike EPAM, it has, moreover, some basic operations such as concept value update, deleting a feature from a node’s description, etc [1]. In addition, UNIMEM transcends EPAM with the weight value of features, providing more realistic classification and prediction for new instances. In EPAM, all or none approach is adopted as mentioned earlier.

There are many other similar hierarchical concept learning methods, which are mostly extensions of EPAM and UNIMEM. For the sake of space, a discussion of these methods is provided in Appendix A.

2.3.2 Web of Concepts

Although the concepts are connected to each other with *is-a* relations, as stated above, numerous neuroimaging and modeling studies reveal that the concepts reside as a highly-connected web in human brain. Nowadays, the widely-accepted belief is the highly-connected functional webs in the brain, activated together to represent a concept [15, 37, 38, 39, 40]. The first proposal of this belief owes its existence to Wernicke and Meynert (see [41, 42] for discussion). According to their proposal, concepts are modality-specific memory located in sensory or motor cortices. Due to its almost fully-connected structure, any clue related with the concept results in the activation of the whole web, and brings the correlated holistic knowledge into the mind.

Goldberg et al. [38] proposed that different information and knowledge activates different places in human brain. For example, tactile information activates the somatosensory, motor, and premotor cortices, whereas taste-related knowledge activates orbitofrontal region which is responsible for decision making and expecting rewards or punishments. Moreover, visual and auditory information cause some reactions in ventral temporal cortex, and superior temporal sulcus, respectively. Kellenbach et al. [39] also justified the proposal with the findings of their experiments for color, size and sound judgments. In his famous work, Pulvermuller [15, 37] stated that premotor and motor actions that are heard during spoken language directly activates the corresponding areas in the brain. For instance, the word “lick” activates the tongue-related area, “pick” affects the finger-area, while “kick” activates the foot-area of the brain. This easily demonstrates that the category-dependent motor actions result in a systematic activation of motor and premotor areas. Another study by Chao and Martin [43] supports this belief. In a tool naming and viewing task, the grasping tool for using it is an integral part of tool concept and activates the ventral premotor area (responsible for hand actions), as well as left posterior parietal cortex (responsible for producing planned movements), eliciting the spatial and motor areas are highly correlated with its semantics. In the light of these studies, we can conclude that the conceptualization process is highly distributed system in the brain as a *web*.

Although these studies reveal really important information, it is only the tip of the iceberg [42, 44, 45, 46]. Their common focus is whether a connected web of pri-

mary cortices is sufficient to explain the conceptualization or there is a dedicated area that organize the low-level cortex activations to constitute the corresponding concept. Damasio et al. [40, 47] mention about the existence of a high-level, amodal convergence areas where the timelocked concept activations occur. Lambon Ralph and Patterson and his colleagues [42, 44] have conducted an experiment on the Semantic Dementia (SD) patients showing that, although they lack of semantic knowledge, other cognitive abilities remain intact. For instance, the area representing the “Zebra” category is damaged in the patients. They cannot recognize the given picture of a zebra, but say that it is a “Horse”. Therefore, this damaged area results in the loss of concept meaning but the gathered information is sufficient for the patients to perceive the shown zebra as a horse, be the most similar concept to it in human perception. There is also another type of dementia occurring in Anterior Temporal Lobe (ATL). The patients of this disease can ask the referent of a herd of sheep, although they are healthy for all kinds of cognitive facilities (see [48, 49, 50]). Another important issue with the help of this is whether ATL is a *semantic hub*, connecting the widespread web of concepts into meaningful entities. They discuss that the features that constitute of concepts combined together in nonlinear and complex manner. One supporting example from the study by Ralph [46] is the comparison of single-layered and multi-layered with hidden layer neural networks. The single-layered NNs can classify linear features, which is impossible to classify certain functions. Even if there is one more layer, these functions can be created.

Although the current studies enlightens our understanding of human brain, there are lots of questions that remain unresolved due to complex structure of the brain. Moreover, we can easily say that the conceptualization is a core process of human perception.

2.4 Concept Studies in Robots

There are numerous studies about conceptualization and learning adjective and noun concepts from sensorimotor data of robots [8, 9, 10]. Learning concepts is an inevitable part for human-robot interaction. Conceptualization is studied not only in cognitive robotics but also other research areas. For instance, the learning ability of

noun categories in humans is extensively studied in psychology [51]. However, there is a big gap between the learning of concepts in humans and the learning of concepts in robots.

One of these learning methods is grounding noun categories to the sensorimotor experiences. Yu and Ballard [52] proposed a multi-modal learning system that grounds symbols (words) in visual properties of objects. There are three possible states in this work which are natural language processing, including the extraction of semantic meaning from the signals of spoken words, visual processing, being the process of getting picture frames from video and extracting visual features (color and shape features) from them, and multi-modal integration, the part of integration of different modalities. Sinapov et al. [8] also developed a multi-modal learning method for noun categories using visual, auditory, and proprioceptive sensory modalities. They make the experiments using 100 objects with 20 object categories. In addition, they also reduce the time for prediction of noun category by applying the behaviors with respect to common and distinctive features of objects. The former increases the probability of classification of a given object as a specific category, while the latter decreases. The selection of the behavior is named as *exploratory behavior* [53]. Another important study about exploratory behaviors comes from Chu et al. [54]. In their study, a PR2 robot is employed in order to learn the adjective concepts using the tactile sensors that are implanted to the hand by applying different behaviors. They have attempted to learn the adjective concepts using both static and dynamic learning methods. If the number of objects that are classified as any adjective category is low, the exploratory procedures (EPs) gain importance for correct prediction.

There are some approaches to learning adjectives using visual features. In the work of Petrosino and Gold [55], they have conducted three tasks to learn the adjectives. In the first task, they have generated a set of words to describe the objects with respect to size, color, and distance, and tried to find the commonly used adjectives for different objects. The second task is to create a ground truth data over objects. They have grouped the adjectives with their antonyms except color adjectives (the color of a served object is asked to a human participant, and one-to-one mapping between each object and color is established), and the participants are asked to choose any one of the adjectives that represents best the presented object and label that object with

this adjective. If they cannot classify this object as any one of dichotomy, then this object is left blank for these two adjective pairs. The final task is conducted to decide whether contrasting objects in a single adjective category is easier than the objects in different categories. According to the results, humans have a tendency to compare the objects with respect to the adjectives in the same adjective category, rather than inter-category adjectives. Another approach is to retrieve the basic form of language by using the perceptual information in a humanoid robot [56]. The main aim in this work is setting up a correlation between words and perceptually grounded meanings. In the work of Chella et al. [57], they have developed a system to again associate the words with the sensorimotor experiences.

As in the learning of adjectives, there are studies focusing on the learning of object concepts differing from adjective learning with the requirement of multi-modal classification [58, 59, 60, 61]. Chauhan et al. [62] developed a method that is open-ended, meaning incrementally forming new categories and their names, to create one-to-one connections between object categories and spoken words to provide better categorization using visual and auditory information. Griffith and his friends also proposed a method to learn the object category with applying different behaviors to them [63]. This method provides to learn the object categories, namely container and non-container, performing behaviors to objects, and generalizing the knowledge to find the category of novel objects. The category learning process basically depends on two outcome features after behaviors. First one is the number of timestamps when the object and the block move together, and the second one is the difference of object and block movements, separately. Beyond the works about human psychology, the object categorization and recognition are also studied in autonomous mobile robots. For example, Gorbenko and Popov [64] have proposed a method to recognize the object category while navigating the environment. According to their proposed method, the recognition of objects can be achieved using autonomously generated neural networks and genetic algorithm. For example, they have two neural systems to detect the red square and blue circles. With the help of this self-learning method, they can produce a neural system to recognize the red circles.

2.5 Probabilistic Graphical Models

In our study, we have concepts and co-occurrences of them, giving us a relation between any two concepts. The concepts that are not initially activated can be triggered using this relation. To model this activation process as a web (graph), we used probabilistic graphical models.

Probabilistic graphical models is the representation of probabilistic models using nodes and edges. Each node in graph refers to the variables; and the link (edges) between nodes depends on the applied problem. If the edges are directed, this graph represents a Bayesian Network [65]. However, in our problem the edges cannot be directed, because there is a correlation between nodes in both directions. In other words, our web must represent $p(x_a|x_b)$ as well as $p(x_b|x_a)$; where x_a and x_b are conditional variables representing the concepts in our experiments. This type of undirected graph can be modeled as Markov Random Field [66] (MRF). Since there is no parent or child relation as in the Bayesian Network graphs, all the functions are defined over maximal *cliques*. A *clique* is a set of fully-connected nodes in a graph, and a maximal clique is the set of maximum possible nodes with fully-connected structure. See Figure 2.2

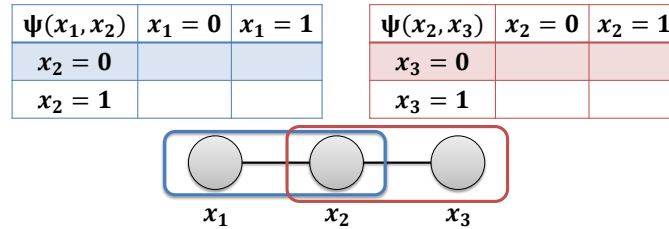


Figure 2.2: Markov random field model with 2 maximal cliques and their potential tables. $\psi(\cdot)$ refers to the clique potential. [Best viewed in color]

Grouping the nodes into maximal cliques enables us to factorize the joint probability distribution over clique nodes. Each clique includes a table consisting of probabilities which are non-negative for its own variables. This table is named as the *potential table*, and can be seen in Figure 2.2.

The joint probability distribution $p(\mathcal{X} = \mathbf{x})$, where \mathbf{x} is a specific configuration of

variables \mathcal{X} , is the normalized product of potentials of cliques ($\psi_c(\mathbf{x}_c)$) and can be formulated as:

$$p(\mathcal{X} = \mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c). \quad (2.1)$$

In addition, the joint probability distribution can be modeled in terms of energy function:

$$p(\omega) = \frac{1}{Z} \exp(-U(\omega)), \quad (2.2)$$

where ω is a possible configuration of concept web W , and $U(\omega)$ is the energy function of the MRF function with a configuration ω , and can be formulated as:

$$U(\omega) = - \sum_{c \in \omega} V_c(c) - \sum_{\mathbf{K} \in \mathcal{K}} V_{\mathbf{K}}(\omega), \quad (2.3)$$

where \mathcal{K} is the set of all possible cliques, c is the set of all activated concepts in a configuration ω , and V_c is the potential of each active concept c , and can be calculated using the distance function ($d(\cdot)$), defined in Equation 3.3:

$$V_c(c) = \exp(-d(\mathbf{x}, c)), \quad (2.4)$$

with \mathbf{x} is the feature vector, extracted on a test object, and c is the active concept. Second term in the energy function is the potential of clique \mathbf{K} and can be defined by:

$$V_{\mathbf{K}}(\omega) = \psi_{\mathbf{K}}(\mathbf{x}_{\mathbf{K}}), \quad (2.5)$$

where $\psi_{\mathbf{K}}(\cdot)$ is the potential of a clique node of variables $\mathbf{x}_{\mathbf{K}}$ (the same term in Equation 2.1). Z is the partition function, in other words the normalizing factor, and can be calculated as:

$$Z = \sum_{\omega \in \Omega} \exp \left(\sum_{\mathbf{K} \in \mathcal{K}} V_{\mathbf{K}}(\omega) \right), \quad (2.6)$$

where Ω is the set of possible configurations.

In our experiments, we have two types of nodes; the separator nodes which are the variable nodes for concepts and represented by square nodes, and the clique nodes created by changing the MRF graph to a maximal clique graph and symbolized with circular nodes. Before explaining belief propagation for our model, we mention about the terms required for understanding the theorem.

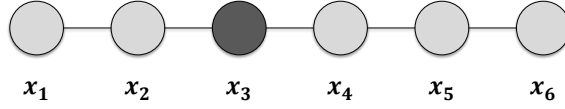


Figure 2.3: Sample Markov Random Field chain consisting of six variables

- **Probabilistic Inference** is the process of finding the posterior distribution of variables, i.e., $p(x_1 = v_1, x_2 = v_2, \dots)$. In our model, the inference is used to find the active concepts in a concept web.
- **Marginal Probability Distribution** is the process of finding the posterior probability of a variable $p(x = v)$, or variables for their values. These variables are named as *query nodes*.

2.5.1 Elimination Algorithm and Belief Propagation

As a term, elimination of variables is an inference method for probabilistic graphical models. For example, we want to find the marginal probability of a variable x_3 in a MRF chain (Figure 2.3). As previously mentioned, marginal probability is calculated by summing all values of the state variables over joint probability distribution:

$$p(x_3) = \frac{1}{Z} \sum_{x_1, x_2, x_4, x_5, x_6} \psi(x_1, x_2) \psi(x_2, x_3) \psi(x_3, x_4) \psi(x_4, x_5) \psi(x_5, x_6). \quad (2.7)$$

As you can see from Equation 2.7, there are unnecessary multiplications. These multiplications increase the computational complexity of the algorithm. For example, the variable x_6 is only on the potential $\psi(x_5, x_6)$. Therefore, we can iterate this summation to the scope of that potential. For all other variables, we can do the same thing. As a result, Equation 2.7 becomes:

$$p(x_3) = \frac{1}{Z} \sum_{x_2} \psi(x_2, x_3) \sum_{x_1} \psi(x_1, x_2) \sum_{x_4} \psi(x_3, x_4) \sum_{x_5} \psi(x_4, x_5) \sum_{x_6} \psi(x_5, x_6). \quad (2.8)$$

After applying the summation over variable x_6 , we obtain the following (see message passing iterations in Figure 2.4):

$$p(x_3) = \frac{1}{Z} \sum_{x_2} \psi(x_2, x_3) \sum_{x_1} \psi(x_1, x_2) \sum_{x_4} \psi(x_3, x_4) \sum_{x_5} \psi(x_4, x_5) m_{x_6 \rightarrow x_5}, \quad (2.9)$$

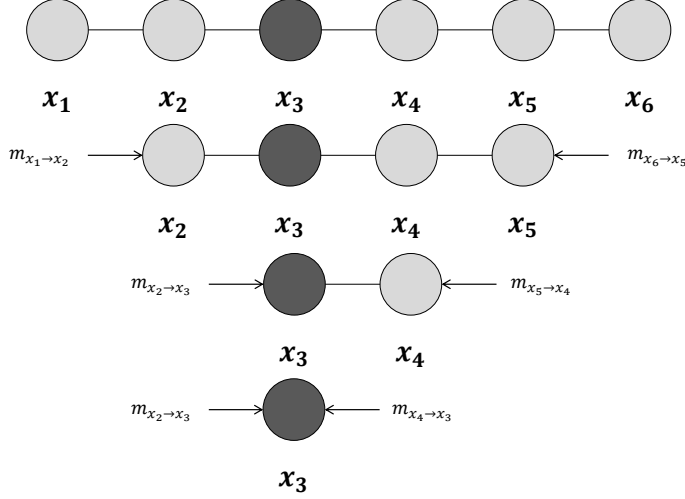


Figure 2.4: Schema for message passing to find the probability $p(x_3)$ for Figure 2.3. (adapted from [2])

where $m(x_5)$ is the intermediate factor after elimination of variable x_6 [2]. We can do the same thing on variable x_5 and Equation 2.9 becomes:

$$p(x_3) = \frac{1}{Z} \sum_{x_2} \psi(x_2, x_3) \sum_{x_1} \psi(x_1, x_2) \sum_{x_4} \psi(x_3, x_4) m_{x_5 \rightarrow x_4}. \quad (2.10)$$

This procedure is applied until obtaining the Equation 2.11 with the elimination order of $x_4 - x_1 - x_2$

$$p(x_3) = \frac{1}{Z} m_{x_2 \rightarrow x_3} m_{x_4 \rightarrow x_3}, \quad (2.11)$$

which gives us the marginal distribution over query node x_3 .

In parallel with the elimination algorithm, belief propagation is used to infer probabilities on graphical models. It can be thought as passing messages from one node to the other one. Therefore, it is named as *message passing*.

In Figure 2.3, the marginal probability distribution is calculated by multiplying the potentials of maximal cliques and dividing by normalization factor as we have described in Equation 2.8.

Instead of elimination, we try to group the variables as left and right hand sides of the query node as in Equation 2.12:

$$p(x_3) = \frac{1}{Z} \left[\sum_{x_2, x_1} \prod_{i=1}^2 \psi(x_i, x_{i+1}) \right] \left[\sum_{x_4, x_5, x_6} \prod_{i=3}^5 \psi(x_i, x_{i+1}) \right]. \quad (2.12)$$

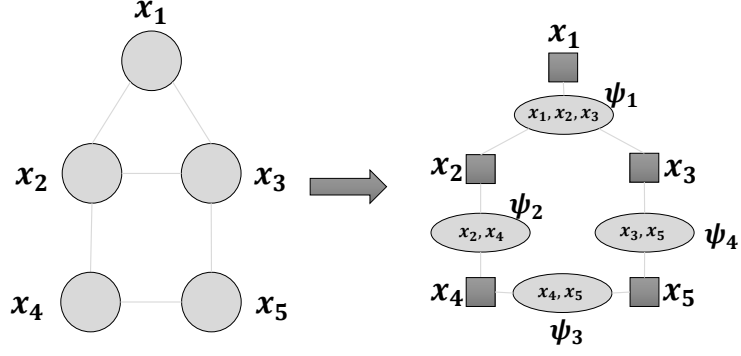


Figure 2.5: MRF graphs and its corresponding graph used in Loopy Belief Propagation. The circular nodes represents the maximal cliques while square nodes shows the variable nodes in initial MRF graph.

The term after the normalization factor is the message passed through all variable nodes which are connected to the query node from left, and the last term is the message passed through right hand side nodes of the query node (Figure 2.4). In other words, they are the message from node x_2 and the message from x_4 , respectively. The final equation becomes:

$$p(x_3) = \frac{1}{Z} \mu_\alpha(x_3) \mu_\beta(x_3), \quad (2.13)$$

where $\mu_\alpha(x)$ is a message from a left node of a node x ; $\mu_\beta(x)$ is a message from a right node of a node x .

2.5.2 Loopy Belief Propagation

Belief propagation algorithm can be applied to tree-structured graphs, such as factor graphs, junction trees as clearly described in the dissertation of Gouws [2] and in Section 2.5.1. These type of graphs are acyclic graphs. Therefore, the message passing algorithm can be applied easily on these graphs. Nevertheless, it is inefficient when compared with standard belief propagation methods.

Another important drawback of the LBP is the several times iteration of message passing. Due to its cyclic structure, message passing algorithm is applied until all values of the variables converge. However, it is possible that the values of variables do not converge. In our case, the highly-connected structure of concepts, however,

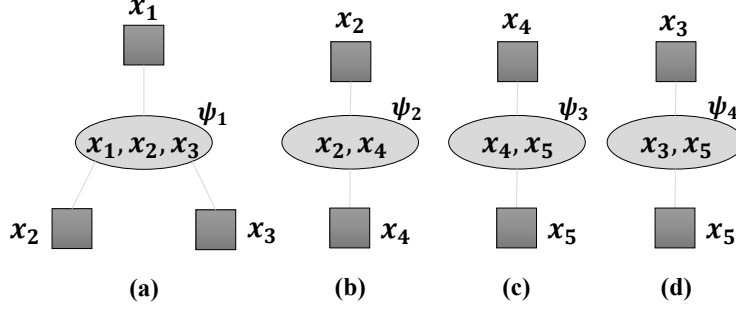


Figure 2.6: Separated cliques nodes with their separator nodes (sub-trees) of Figure 2.5

leads to the convergence of a graph with concepts whose activations are fully activated ($p(x) = 1.0$) or not activated ($p(x) = 0.0$).

As a representation of concepts in a web, the circular nodes represents the maximal cliques, and the square nodes are variables and named as *separator nodes* (Figure 2.5). In this method, each clique node and its separator nodes are tackled separately. These separator nodes can be shared among different clique nodes. Therefore, they must be updated twice or more according to its number of existence in clique nodes. One message sent from clique node to the separator node, named backward pass, is used in an another subtree as a forward pass of message. A backward message is stored in a potential table of a separator node, and named as *separator potential* and symbolized with $\phi(\mathbf{x})$. These separator and clique potentials must be updated in each iteration of message passing.

Initially, we have to set each cell of the separator potential table to one. As in the Figure 2.6, each subtree has one root node (clique) and the leaf nodes (concepts). These leaf nodes send a message to the root node. The message results in an update of clique potentials using Equation 2.14:

$$\psi_{\mathbf{K}}^*(\mathbf{x}_{\mathbf{K}}) = \psi_{\mathbf{K}}(\mathbf{x}_{\mathbf{K}}) \prod_{x_m \in ne(\mathbf{x}_{\mathbf{K}})} \phi_m(x_m), \quad (2.14)$$

where $\psi_{\mathbf{K}}^*(\mathbf{x}_{\mathbf{K}})$ is the updated potential value of $\psi_{\mathbf{K}}(\mathbf{x}_{\mathbf{K}})$ over set of variables $\mathbf{x}_{\mathbf{K}}$ in clique node \mathbf{K} ; x_m is the connected separator node to the clique \mathbf{K} .

After updating the potential table of the clique, the potentials of separator nodes connected to a clique node must be updated with respect to updated potential table of a

clique. Therefore, a clique node sends forward pass messages to separator nodes:

$$\mu_{\mathbf{K}^* \rightarrow x_m}(x_m) = \sum_{\mathbf{x}_n \in \mathbf{x}_{\mathbf{K}} \setminus x_m} \psi_{\mathbf{K}^*}(\mathbf{x}_n). \quad (2.15)$$

As previously explained, the separator node potentials are the storage of messages passing towards another subtree. Therefore, we have to update the potential with the message from updated clique node \mathbf{K}^* . If there is an old message in potential table, we have to divide the new message with the previous one, denoted by $\mu_{\mathbf{K} \rightarrow x_m}$ and multiply with the old separator potential as in Equation 2.16:

$$\phi_m^*(x_m) = \phi_m(x_m) \times \frac{\mu_{\mathbf{K}^* \rightarrow x_m}(x_m)}{\mu_{\mathbf{K} \rightarrow x_m}(x_m)}. \quad (2.16)$$

Otherwise, we directly update the potential of a variable by multiplying the previous potential value with the message from the clique node \mathbf{K} to the separator node (variable) x_m :

$$\phi_m^*(x_m) = \phi_m(x_m) \times \mu_{\mathbf{K}^* \rightarrow x_m}(x_m). \quad (2.17)$$

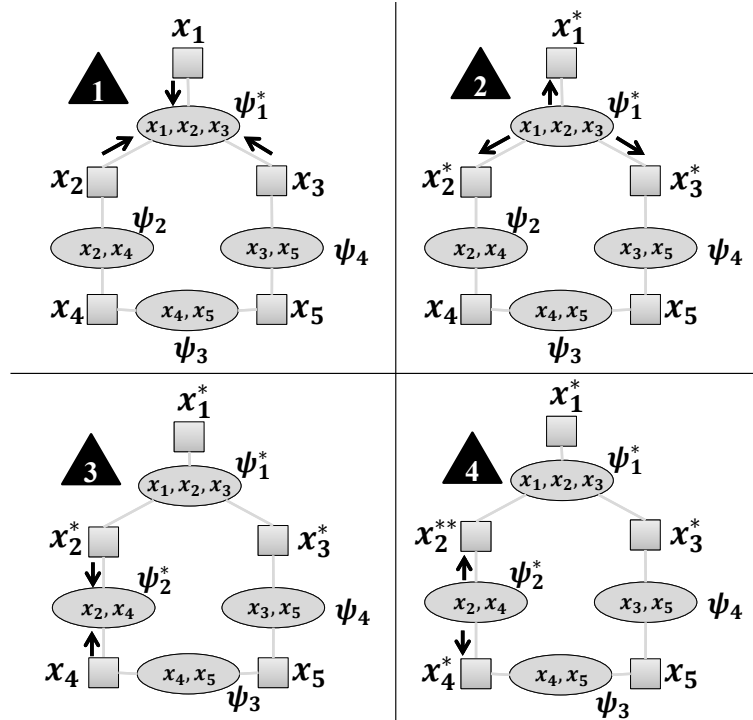


Figure 2.7: Sample initial four iteration for loopy belief propagation in Figure 2.5. Double star means that the value of potential table is updated twice.

For loopy belief propagation algorithm, we iterate Equations 2.14, 2.15, and 2.16 step-by-step for each clique node, in other words, each subtree until the potential tables converges or some threshold iteration number is exceeded. Sample execution of this algorithm for the factor graph depicted in Figure 2.5 can be seen in Figure 2.7.

Finally, we find the posterior probability $p(x_n)$ for any variable x_n using any clique containing x_n by marginalizing the factor potentials over clique variables except x_n

$$p(x_n) = \sum_{x_l \in \mathbf{x}_{\mathbf{K}} \setminus x_n} \psi_{\mathbf{K}}(x_l). \quad (2.18)$$

2.6 Support Vector Machine (SVM)

The basic definition of SVM learning is clustering the multi-dimensional feature vectors into any number of clusters by dividing the space with multi-dimensional hyperplane equation [67] (Figure 2.8). In the famous book of Vapnik [68], he used Support Vector Machine for classification and regression analysis. It is commonly used technique in concept learning. In our study, we have used libSVM [69] and WEKA Open-Source Data Mining Software [70].

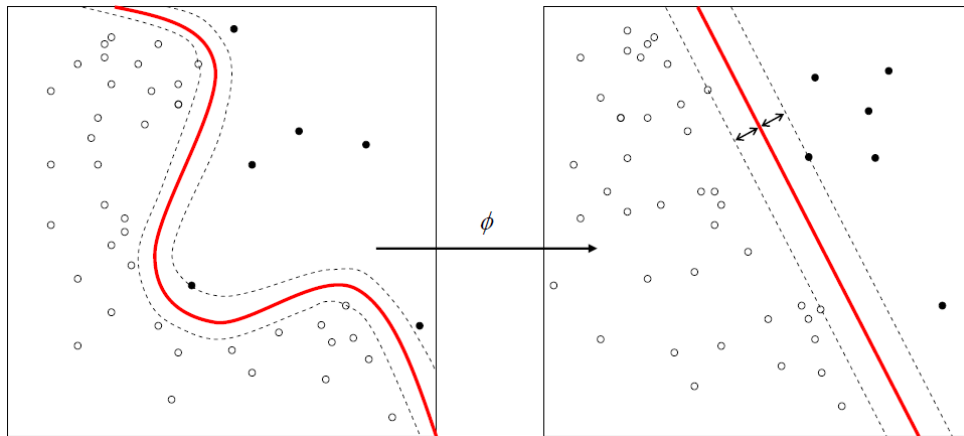


Figure 2.8: Schema for dividing multi-dimensional feature space into two clusters with hyperplane using kernel functions of SVM. (Figure from [3])

CHAPTER 3

EXPERIMENTAL SETUP AND CONCEPTUALIZATION

In this chapter, we mention about the experimental setup and the methods that are used for concept learning and prediction of categories.

3.1 Experimental Setup

In this section, the hardware and software components are described.

3.1.1 Hardware Components

3.1.1.1 iCub Humanoid Robot Platform

iCub [71] is an open-source robot platform that was developed within the EU project RobotCub and currently being used in many research laboratories all around the world. It is commonly being used in cognitive robotics. As a physical structure, it seems like a 3.5 years old child. It has in total 53 joints; six joints for head, 16 joints for each arm, three joints for torso, and six joints for each leg. Moreover, it has sensors to perceive the environment. Some of these sensors are microphone, cameras, tactile sensors on each fingertip (Figure 3.1). We frequently use these tactile sensors to arrange the grasping pressure of the robot hands on an underlying object. Beyond these, it also provides information about the property of an object. We also use the microphone to decide whether an object has an internal sound. Besides, iCub has cameras inside the eyeball. We can gather depth information using these cameras.

Due to calibration problems, we have decided to use the Kinect sensor.

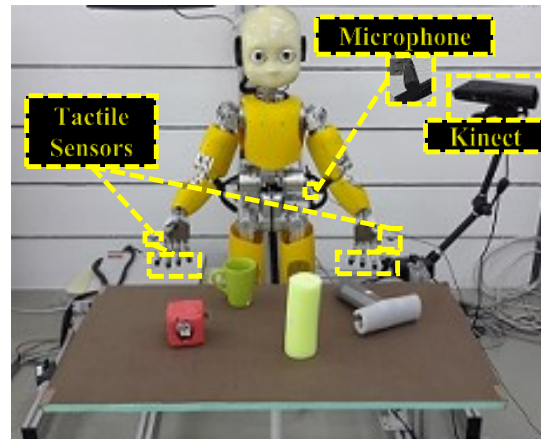


Figure 3.1: iCub Humanoid Robot Platform with microphone and Kinect sensor (Figure taken from [4]).

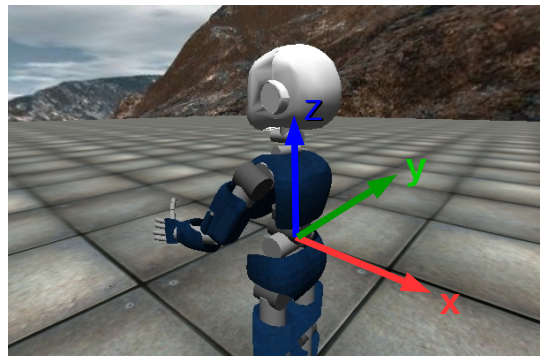


Figure 3.2: iCub Humanoid Robot Platform coordinate reference frame.

3.1.1.2 Kinect

Kinect (Figure 3.3) is manufactured by Microsoft to capture motions of players who play Xbox Gaming console. It has tilting motor, microphone, and RGB camera and IR emitter. In addition to RGB colored image, it also provides depth information of the surroundings. It is comparably precise and cheaper than any other depth camera. It takes 30 frames per second, each of which has 640x480 resolution. The range of the depth is between 1 and 4 meters away from the focal point.



Figure 3.3: The Kinect sensor

3.1.2 Software Components

3.1.2.1 iCub Modules

iCub modules provide a robust API to send commands to iCub joint encoders. Instead of sending angle values to joints, we can also send 3-dimensional target position in the reference frame of iCub (Figure 3.2) using inverse kinematic modules, which converts the trajectory of motion from initial position to the target position into the set of joint values to provide motion-safe behaviors.

3.1.2.2 Yet Another Robot Platform (YARP)

YARP [72] is an open-source library developed in order to operate the humanoid robot platforms. It is basically used to provide communication framework for the dependent or independent software modules using network protocols. Initially, it is created for iCub and, later on, developed for all types of modules that require inter-module communication.

3.1.2.3 Point Cloud Library (PCL)

PCL [73] is a robust and computationally powerful library for 3D point cloud data and its geometrical calculations. With the help of this library, we can easily find the geometric properties of a 3D surface. For example, normal vectors of each point in cloud can be calculated easily using normal estimation module. Moreover, there are

other libraries integrated in PCL to provide better thread operations, pointer types and computation of matrix, vector operations, namely Boost and Eigen [74] libraries.

3.1.2.4 Ubigraph

Ubigraph [5] is an open-source library to visualize the dynamic graphs. It provides really efficient and fast creation of concept nodes in our experiments. We use this library to show the activated concepts in the concept web.

3.2 Perception

Perception of the environment and objects is really important for our experiments. We perceive the world with three devices, namely Kinect, microphone and tactile sensors.

3.2.1 Features

One of the most important milestones before starting experiments is to decide which features are relevant for our work. The quality of our experiments is directly affected by the quality of features. If we add too many irrelevant features in addition to relevant ones, the results can be deteriorated. On the other hand, if we select less features than necessary, the features may include insufficient information for learning. The total set of the features can be seen in Table 3.1. We have different number of features for each modality. For this thesis, we have visual, audio, proprioceptive and haptic modalities.

Visual features are extracted using PCL modules [73]. They are really important to identify the corresponding noun and adjective categories of an object. Therefore, the orientation of an object is adjusted with respect to its characteristic properties. For instance, in order to discriminate a cup from a cylinder, it has to be placed on the table with an appropriate orientation such that the handle of a cup can be clearly discerned. The first six visual features are used to get the position of an object and dimension information. The successor feature, namely *object presence*, is set to one

Table 3.1: Features extracted from the interactions with the environment. Parenthesized numbers indicate the index of features in the feature vector.

Feature Type	Feature	Position
Visual	position: (x, y, z)	1-3
	object dimensions: $(width, height, depth)$	4-6
	object presence: $(1, -1)$	7
	normal zenith histogram bins	8-27
	normal azimuth histogram bins	28-47
	shape index histogram bins	48-67
Audio	13 bins of MFCC result (max - min)	68-80
Haptic	Change for index finger	81
	Min values for index finger	82
	Max values for index finger	83
	Mean for index finger	84
	Variance for index finger	85
	Standard deviation for index finger	86
Proprioceptive	Change for index finger	87
	Min values for index finger	88
	Max values for index finger	89
	Mean for index finger	90
	Variance for index finger	91
	Standard deviation for index finger	92

if an object is on the table. The normal vectors are calculated using normal estimation module in PCL. This module calculates the normal vectors for each point in the cloud by specifying the radius of neighborhood. After finding the normal vectors, we find zenith and azimuth values of each normal vector and put each zenith and azimuth normal vectors into histogram bins, separately. Each histogram consists of 20 bins. The remaining visual features comes from the principal curvatures. The maximum and minimum principal curvatures are calculated by using PCL principal curvature estimation module. After finding the maximum and minimum principal curvatures for each point in a cloud, we find the shape index values for these points by using the work studied by Koenderink and Von Doorn [75].

Another important feature group is used to determine whether an object is noisy or silent. For audio features, we use a 13-bin histogram created by Mel-Frequency Cepstral Coefficients (MFCC [76]). Although MFCC is widely applied for speech recognition [77], we manipulated the use of MFCC according for our needs. For each behavior, we collect audio data starting from the beginning up to the completion. By

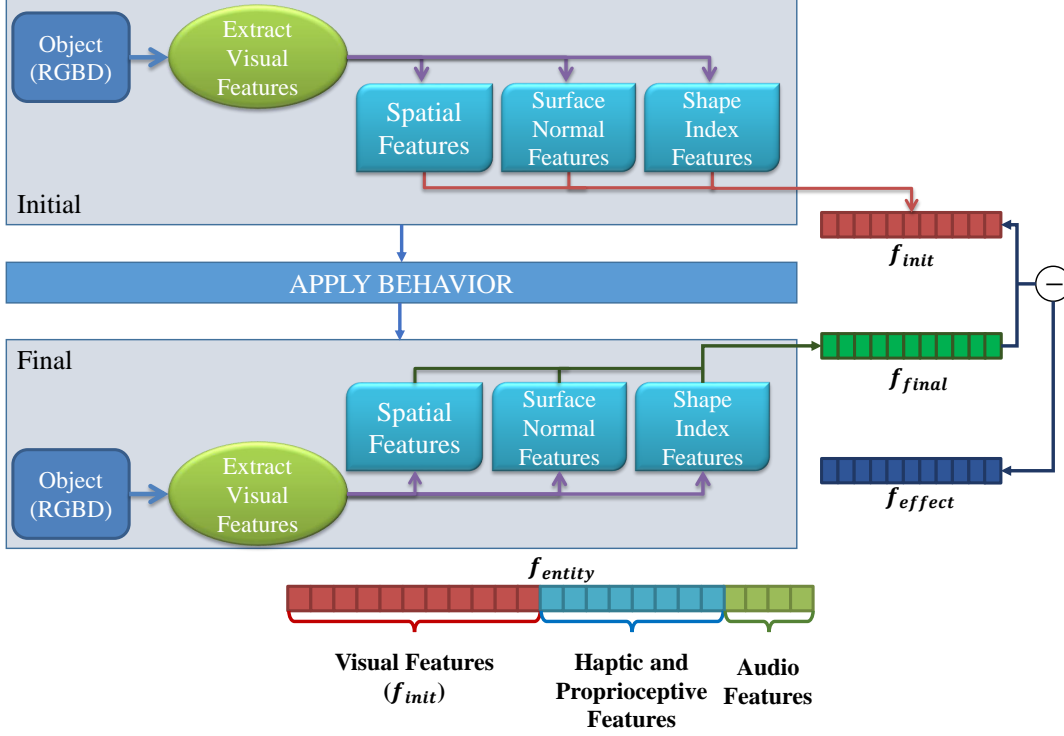


Figure 3.4: Content of effect and entity features

using the collected audio data, we employ the MFCC algorithm to obtain a set of 13-bin features. The number of features in a set changes with respect to the duration of a behavior. After finding a set of MFCC results, we find the maximum and minimum values for each MFCC column (bin) in a set, and subtract them from each other. Therefore, we obtain 13 features for a specific audio file.

Haptic features are directly related with the *grasp* behavior. While grasping an object, the haptic values for index finger are stored in order to determine the hardness of an object with using proprioceptive features, cooperatively. iCub has 12 pressure sensors for the index finger. For calculating the haptic features, we get a maximum valued sensor reading. After getting a number of haptic values, we apply some statistical operations to them, such as mean, variance, standard deviation, as well as, change of maximum and minimum haptic values.

Proprioceptive features are computed like haptic features. We only get encoder values for the index finger during a behavior, and apply the same set of operations described in haptic features.

For different conceptualization processes, we need different set of features. Therefore, we have used two sets of features in the experiments. These sets are named as entity and effect feature sets. The former is directly related with the internal properties of an object, such as hardness, thickness, etc, whereas, the latter gives information about the applied behavior to an object.

Entity features consist of visual, audio, proprioceptive and tactile information and are obtained from an *Grasp-Shake* behavior. The reason of this is that the complete set of features including different modalities can be obtained in one behavior (*Grasp-Shake*).

On the other hand, the effect features are necessary for detecting the behavior and the effect outcome on an object. Therefore, they must include the information about the behavior. We get the visual features before and after applying a behavior to an object, and subtract them. As a result, we obtain the change of the visual properties of an object for an applied behavior. For instance, if we apply *push-left* behavior, all of visual features become zero except the feature related with y-position. You can see the contents of effect and entity features in Figure 3.4.



Figure 3.5: All objects which are separated according to their noun categories. [Best viewed in color]

Table 3.2: The possible effect outcomes of behaviors on different noun categories. Empty cells imply that these behaviors are not applied to the objects in the category. (*arg*: *Left, Right, Forward, Backward*)

	grasp	drop	throw	knock down	push <i>arg</i>	grasp & move <i>arg</i>
box	grasped	moved left disappeared	moved forward disappeared	knocked down moved right disappeared	moved <i>arg</i>	moved <i>arg</i>
ball	grasped	moved left disappeared	moved forward disappeared	knocked down moved right disappeared	moved <i>arg</i> disappeared	moved <i>arg</i>
cup	grasped	-	-	-	moved <i>arg</i>	moved <i>arg</i>
cylinder	grasped	moved left disappeared	moved forward disappeared	knocked down moved right disappeared	moved <i>arg</i> disappeared	moved <i>arg</i>
plate	grasped	-	-	-	moved <i>arg</i>	moved <i>arg</i>
tool	grasped	moved left	moved forward	knocked down moved right	moved <i>arg</i>	moved <i>arg</i>

3.3 Data Collection

We used as many objects as possible. The number of objects used in experiments is 60 (Figure 3.5). Moreover, we divided this set as a training and testing set arbitrarily with cardinalities 45, and 15, respectively. We labeled these objects with respect to noun and adjective categories. We have in total six noun categories $\mathcal{N} = \{box, ball, cup, cylinder, plate, tool\}$, and 10 adjective categories $\mathcal{A} = \{hard - soft, noisy - silent, tall - short, thin - thick, round - edgy\}$.

The adjective set in fact includes the combination of adjective pairs unlike the noun set. These adjective pairs consist of an adjective and its antonym. This dichotomy provides convenience to predict the adjective category of an object, which will be explained in the following sections.

The repertoire for behaviors is kept wide. We have in total 13 behaviors, namely *grasp, drop, throw, knock down, grasp & shake, push left, push right, push forward, pull backward, grasp & move left, grasp & move right, grasp & move forward, grasp & move backward*. iCub does not apply these behaviors to each object. The appli-

cability of these behaviors changes with respect to the noun category of an object, which will be explained in Chapter 5.

After applying possible behaviors to each object, we label the operations with *success* or *failure*. Failure means that the applied behavior either cannot affect an object or does not end up with the predicted outcome for this behavior. If you command the robot to grasp an object that the robot cannot do, then it will eventually fail and the final visual features will remain almost the same as in the initially gathered features. This exemplifies the first condition. The second condition occurs if we apply *knock down* behavior to an object such as a ball whose height is small. The effect of the behavior on a ball-like object may be the same as the effect of applying *push right* behavior to it. Therefore, the resultant effect for both *push right* and *knock down* may be *moved right*. The effects of a behavior on different objects changes with respect to the noun categories of them. In our experiments, the possible effect outcomes of the behaviors on different noun categories can be seen in Table 3.2.

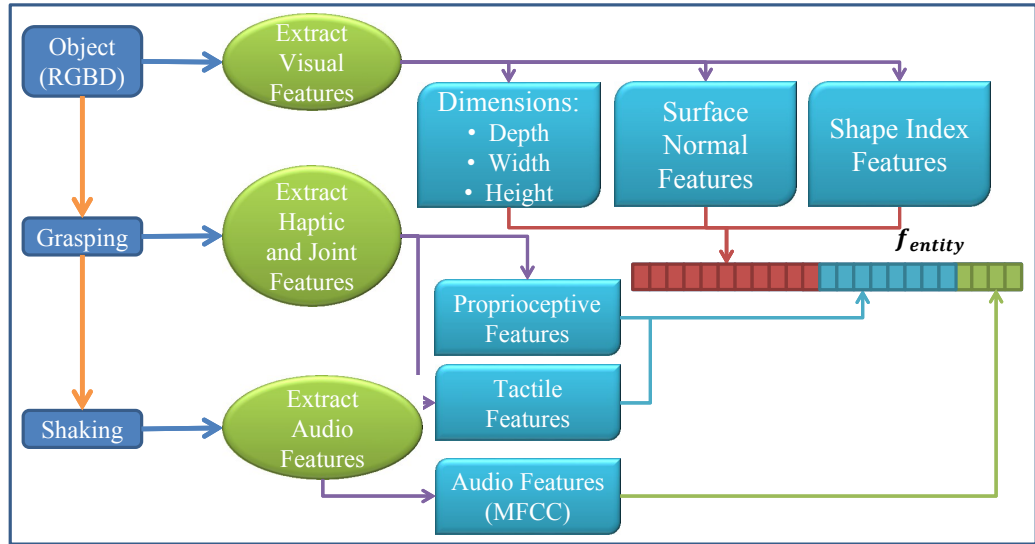


Figure 3.6: Entity features extraction applying *grasp* & *shake* behavior in order of data collection procedure

3.4 Feature Extraction

As we have previously mentioned, we have two types of sets for features. Entity features gives clues about an object properties, while effect features are used to predict the behavior applied to an object. Entity features are obtained by applying *grasp-shake* behavior since some features can only be extracted by grasping (haptic and proprioceptive features) and shaking (auditory features) an object. You can see feature extraction in order of occurrence in Figure 3.6. The grasping behavior directly affects the borders of hard and soft categories. Therefore, we have developed a very effective grasp algorithm in order to get the best results even if tactile value or orientation of the object during behavior changes. We only get necessary tactile information using the index finger. The other fingers are used to grasp an object correctly and provide better sensor reading.

Algorithm 1: Derivation of Prototypes. Algorithm from [7]

for all l in the set of categories \mathcal{C} **do**

- Compute the mean ${}_i\mu_l$ for each feature i :

$${}_i\mu_l = \frac{1}{N} \sum_{e \in l} {}_ie, \quad (3.1)$$

where N is the cardinality of the set $\{e | e \in l\}$; and ${}_ie$ is the i^{th} value of vector \mathbf{e} .

- Compute the variance ${}_i\sigma_l$ of each feature dimension i :

$${}_i\sigma_l = \frac{1}{N} \sum_{e \in l} ({}_ie - {}_i\mu_l)^2. \quad (3.2)$$

end for

- Apply Robust Neural Growing Gas (RGNG) algorithm [76] in the space of $\mu \times \sigma$.

if Effect prototypes are being extracted **then**

- Manually assign the labels '+', '-', '*', and '0' to four clusters that emerge in the previous step.

else

- Manually assign the labels '+', '-', and '*' to three clusters that emerge in the previous step.

end if

3.5 Conceptualization of the Categories

We adopt the prototype-based conceptualization method for noun, adjective, and behavior categories [78]. Prototype-based conceptualization provides us representative

features for any category, so that only relevant features are used to determine the category of a concept, precisely. There are four signs for features. They have different meanings. The feature labeled as ‘+’ is important or relevant for a category, meaning that the values of this feature reside in small interval and relatively larger when compared with other categories. On the other hand, ‘-’ labeled features are also relevant and representative for a category. However, these features have a small variance and mean. If any feature in a category fluctuates largely, then this feature cannot show the general characteristic of a category, and so, is considered inconsistent. These features are labeled as ‘*’. These signs are used in all noun, adjective, and behavior categories. However, the last sign is used only in behavior and effect prototypes. ‘0’ sign means that there is no noticeable change before and after behavior for that feature. The reason why we use this sign only in behavior prototypes is that it can only show the change for a feature.

Robust Growing Neural Gas (RGNG) method is used to find the clusters of these signs. The robustness comes from unsupervised nature in separating the clusters. The application of this method for our purposes can be seen in Algorithm 1.

3.6 Category prediction

After extracting the prototypes for each category, now, we can predict the adjective and noun categories of testing objects. To find the categories of a testing object, we have to decide which categories conform with the object, in other words, which prototypes of the categories best describe the features of a testing object. Therefore, we have to find the closeness of an object to a category. We find the distance between an object and the categories using Euclidean distance formula:

$$d(\mathbf{x}, c) = \sqrt{\sum_{i \in R(\mathcal{C}) \setminus R^*(\mathcal{C})} ({}_i\mathbf{x} - {}_i\mu_c)^2}, \quad (3.3)$$

where \mathbf{x} is the feature vector of an instance; \mathcal{C} is the prototype of category c ; $R(\mathcal{C}) \setminus R^*(\mathcal{C})$ is the set of relevant dimensions in \mathcal{C} ; ${}_i\mathbf{v}$ is the i^{th} value of a vector \mathbf{v} ; and μ_c is the mean values of the features of all instances classified as category c .

3.6.1 Prediction of Noun Categories

Euclidean distance gives us the distance between category and testing object features. However, we want to find the most suitable category for a testing object with probability values. Therefore, we developed a moment-like method in physics to find the probabilities. The smaller the distance between features of an object and a noun category, the more portion this distance takes from the total probability (1.0). This can be formulated as follows:

$$s_{perc}^n(\mathbf{x}, n_p) = \frac{\prod_{n_1 \in \mathcal{N} \setminus n_p} d(\mathbf{x}, n_1)}{\sum_{n_1 \in \mathcal{N}} (\prod_{n_2 \in \mathcal{N} \setminus n_1} d(\mathbf{x}, n_2))}, \quad (3.4)$$

where n_p is the predicted noun category.

3.6.2 Prediction of Adjective Categories

The prediction of adjective categories can be thought as selecting the most suitable adjective category from adjective pairs. In noun category prediction, we have four noun categories, and the algorithm must select any one of them. For this time, we have only two adjectives, one of which must be determined. Therefore, the classification of a test object as *hard* or *soft* directly depends on the distances between entity features of an object and mean features of all instances in these categories separately. Although the characteristic property of Equation 3.4 remains the same, it is simplified to choose the best representative adjective category for a given testing object:

$$s_{perc}^a(\mathbf{x}, a_p) = \frac{d(\mathbf{x}, \bar{a}_p)}{d(\mathbf{x}, \bar{a}_p) + d(\mathbf{x}, a_p)}. \quad (3.5)$$

where \bar{a} is the antonym of adjective a .

3.6.3 Prediction of Verb Categories

Verb categories are predicted using effect features collected after the application of behaviors. The effect feature sets, as well as the verb prototypes, are comprised of only visual features, as mentioned before. As in the prediction of noun categories, we have more than two category options to predict. Therefore, we apply almost the same

formulation described in Section 3.6.1:

$$s_{perc}^v(\mathbf{x}_{vis}, v_p) = \frac{\prod_{v_1 \in \mathcal{V} \setminus v} d(\mathbf{x}_{vis}, v_p)}{\sum_{v_1 \in \mathcal{V}} (\prod_{v_2 \in \mathcal{V} \setminus v_1} d(\mathbf{x}_{vis}, v_p))}, \quad (3.6)$$

where \mathbf{x}_{vis} is the only visual features of an object; v_p is the verb category.

3.6.4 Prediction of Effects Using SVM

Additionally, the system is expected to be able to guess the effect label of a behavior on any given object. We have trained separate SVMs for each behavior using the given entity feature vectors from the training set, and their corresponding effects. The effects on the training set objects are labeled by hand ¹ (The expected effects for each noun category is shown in Table 3.2). At the end of training, given the entity features of an unknown object, the system is able to predict the effects for each behavior on this object. A similar scheme for making sense of behaviors has been utilized in [7]. Eventually we obtain confidence values of effects for testing objects using the trained SVMs.

¹ This can be considered as an implementation of the influence of language on concept formation. See Section 5.1.1 for a short discussion on this influence

CHAPTER 4

EFFECT OF CO-OCCURRENCE ON CATEGORY PREDICTION

In this chapter, the effect of co-occurrence on category prediction is analyzed.

4.1 Contribution of co-occurrence information

The adjective and noun concepts are learned separately using perceptual similarity. However, we want to see whether co-occurrence information between adjectives and nouns enhances the predictions. Therefore, we developed a method to predict the categories of an object. While predicting the adjective categories, the co-occurrence information between noun category of an object and adjectives are used and vice versa. Although co-occurrence is used in many applications such as cross-situational learning of words and objects by Yu and Smith [79], it is not efficiently used in concept learning except [80].

Predicting a noun category using co-occurrence information is a two-level process; the first phase is the prediction of a noun category using perceptual similarity, and the second one is addition of co-occurrence information between an adjective and a noun category, which is calculated with the following formula:

$$c_n(n_p, \hat{\mathcal{A}}_x) = \sum_{a \in \hat{\mathcal{A}}_x} \frac{c(n, a)}{|\hat{\mathcal{A}}_x|}, \quad (4.1)$$

where $\hat{\mathcal{A}}_x$ is the set of predicted adjectives using perceptual similarity; $|\mathcal{S}|$ is the cardinality of set \mathcal{S} ; $c(n, a)$ is the co-occurrence value of noun n with adjective a considering only relevant (non-‘*’ entries of Table 4.1).

Table 4.1: Prototypes of noun and adjective co-occurrences. ‘*’, ‘+’ and ‘-’ respectively represent inconsistent co-occurrence, consistent co-occurrence and consistent non-co-occurrences.

Noun	Hard	Soft	Noisy	Silent	Tall	Short	Thin	Thick	Round	Edgy
Box	+	-	*	*	-	+	-	+	-	+
Cylinder	+	-	*	*	*	*	*	*	+	-
Cup	+	-	+	-	-	+	-	+	+	-
Ball	*	*	-	+	-	+	-	+	+	-

For combining these two prediction methods namely, perceptual similarity and co-occurrence, we developed a method where the contributions of each can be weighted using weighting constant (see Equation 4.2).

$$s_{comb}^n(\mathbf{x}, n_p) = (1 - \omega_{an}) \times s_{perc}^n(\mathbf{x}, n_p) + \omega_{an} \times c_n(n_p, \hat{\mathcal{A}}_x), \quad (4.2)$$

where $\omega_{an} \in [0, 1]$ is the weight which controls the contribution of the prediction from adjectives.

Almost the same procedure is applied to predict the adjective categories:

$$s_{comb}^a(\mathbf{x}, a_p) = (1 - \omega_{na}) \times s_{perc}^a(\mathbf{x}, a_p) + \omega_{na} \times c(n_{\mathbf{x}}, a), \quad (4.3)$$

where $\omega_{na} \in [0, 1]$ is the weight controlling the contribution of prediction from the nouns; $n_{\mathbf{x}} \in \mathcal{N}$ is the predicted noun category from features.

4.2 Cross-Situational Labeling of Categories

Labeling concepts with appropriate English words is an inevitable part of concept learning in humanoid robots, since the robot must learn the corresponding words of concepts in order to perform a reliable Human-Robot Interaction.

Firstly, we label adjective concepts with adjective concept names, such as a_1, a_2, \dots , and noun concepts with noun concept names n_1, n_2, n_3, \dots . We have in total 10 adjectives and four noun categories so we have totally 14 concept names. In our training dataset, and we have a row of object name, noun category and a random number of adjective labels:


```

obj1, box, noisy, tall, edgy
obj2, cup, short, round
obj2, cup, noisy, short, round, thick, hard
...
objn, ball, round

```

All of these rows are created with respect to random selection of the combinations of the supervised adjective labels. In other words, there can be more than one row showing the same object and its properties, and the properties can be a whole set of possible adjectives; noun category of an object or a subset of them. For example, if we choose 60%, this means that we only use the 40% of the all possible row combinations.

After creating the dataset, we apply two different algorithms in order to label the concepts, namely cross-situational labeling and its modified version (Appendix B). To show the enhancements, we compared the modified and default version of cross-situational labeling process.

4.3 Results

We have 20 objects for each training and testing. Moreover, we have used only four noun categories $\mathcal{N}_{cooc} = \mathcal{N} \setminus \{plate, tool\}$ in this experiment (See Figure 3.5 for all noun categories) while the set of adjective categories remains the same $\mathcal{A} = \{hard - soft, noisy - silent, tall - short, thin - thick, round - edgy\}$. Our comparison criteria is mainly to show the enhanced prediction accuracies using co-occurrence information. After showing the experimental results, we demonstrate a game-like application, namely “What object is it?”, to predict the most probable noun categories of a given set of adjectives. After that we will show the accuracy for concept labeling using modified cross-situational labeling method (Algorithm 2).

4.3.1 Noun and Adjective Prediction using SVM

As we have previously mentioned in Chapter 2, SVM is a widely-used machine learning method for separating dataset into a number of cluster. We have trained SVM to learn noun concepts ($\mathbf{x} \rightarrow \mathcal{N}_{cooc}$), and adjective concepts for each adjective dichotomy ($\forall \mathcal{A}_x \in \mathcal{A}_{pair}, \mathbf{x} \rightarrow \mathcal{A}_x$). For each training, we have used 5-fold cross-validation.

As you can see from Table 4.2, training accuracies for both noun and adjective categories using *only* perceptual similarity (s_{perc}) is better than using SVM. Another important outcome is that noun categories are learned better than adjective categories, which justifies the study of Gasser and Smith [81].

Table 4.2: Average noun and adjective prediction accuracy results on the **training** set.

	Perceptual Similarity (s_{perc})	SVM
Nouns	100%	90%
Adjectives	94%	88%

4.3.2 Co-occurrence Effect on Prediction

It has been previously shown that learning adjectives are more difficult than nouns ¹. Due to this property of adjectives, there may be more wrong predictions in adjective categories than noun categories. However, we postulate that co-occurrence information between adjectives and nouns can correct the wrong predictions for both noun and adjective categories. To show the effect of co-occurrence, we have increased the co-occurrence weight ω_{na} (Equation 4.3). As it can be seen in Figure 4.1, the accuracy for correct prediction accrues as the weight constant increases.

In our case, using only perceptual information is sufficient to predict the noun categories of the testing objects. The possible reason of this is the hypothesis of claiming that nouns are learned easier than adjectives. To show the effect of co-occurrence information, we have added 40% noise to the noun prediction. Hence, the percentage of correctly predicted noun categories for testing objects initially starts with 65% . As in

¹ According to this hypothesis described in Psychology [82] and Linguistic [83]; the adjective categories depend on less features when compared with noun categories. Therefore, it is difficult to discover the relevant dimensions in multi-dimensional feature space for adjectives. We have also justified this hypothesis in our experiment [80].

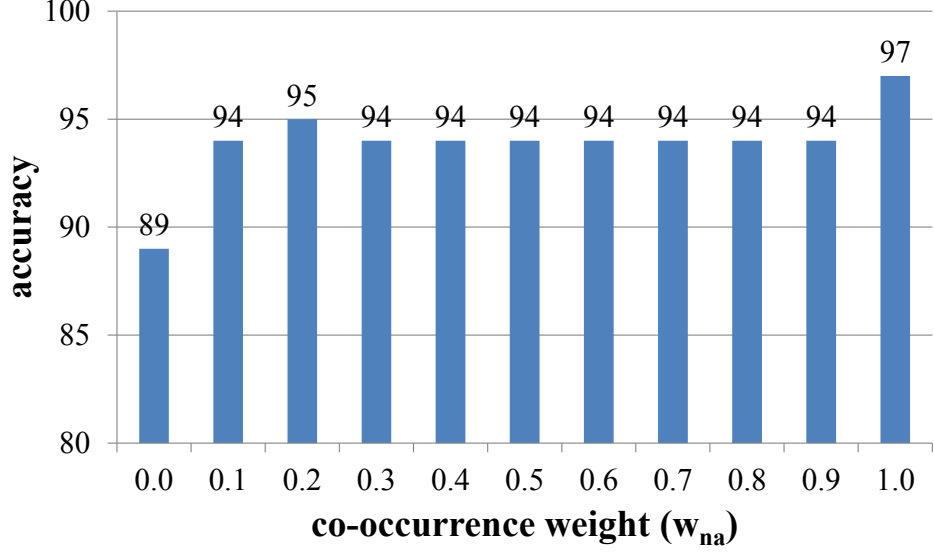


Figure 4.1: Adjective prediction accuracy for the testing set with respect to weighted contribution of co-occurrence

the previous case, we increase the co-occurrence weight for noun ω_{an} (Equation 4.2). Although the percentage decreases after 0.4, it reaches to a peak value for $\omega_{an} = 1.0$ in Figure 4.2.

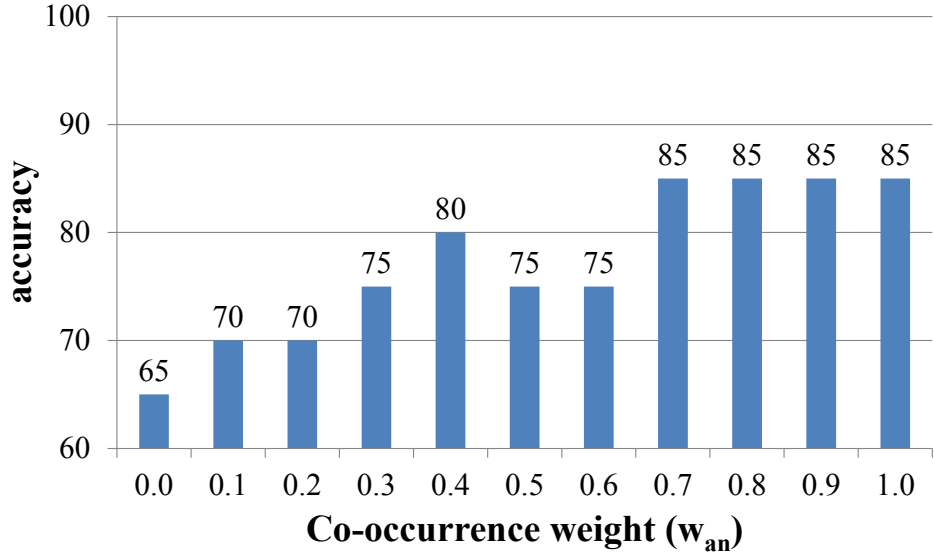







Figure 4.2: Noun prediction accuracy for the testing set with respect to weighted contribution of co-occurrence

After giving the prediction accuracies of noun and adjective categories for test objects, we want to give object specific results in order to show the corrected prediction over

adjective and noun categories.






Table 4.3: Predicted adjectives for some objects from the test set (bold denotes correct classification). The co-occurrence weight w_{na} is taken as 0.2 where prediction performance is maximized.

Objects	Adjectives		
	Perceptual Similarity (s_{perc}^a)	Perc. Similarity and Co-occurrence s_{comb}^a	SVM
 O_1	hard (61%) noisy (67%) tall (54%) thick (55%) round (54%)	hard (65%) noisy (70%) short (51%) thick (60%) round (59%)	hard (95%) noisy (93%) short (92%) thick (74%) round (76%)
 O_2	hard (55%) silent (67%) tall (64%) thin (54%) edgy (55%)	hard (59%) silent (66%) tall (57%) thick (51%) edgy (60%)	hard (75%) silent (89%) short (69%) thin (54%) round (59%)
 O_3	hard (54%) silent (61%) short (56%) thick (53%) edgy (57%)	hard (59%) silent (60%) short (60%) thick (58%) edgy (61%)	hard (82%) silent (89%) short (92%) thick (96%) edgy (89%)
 O_4	soft (60%) silent (58%) short (52%) thick (53%) round (54%)	soft (59%) silent (63%) short (57%) thick (57%) round (59%)	soft (99%) silent (93%) short (88%) thick (54%) round (98%)
 O_5	hard (56%) silent (73%) short (53%) thin (51%) round (52%)	hard (61%) silent (70%) short (53%) thin (51%) round (56%)	hard (73%) silent (83%) short (78%) thick (62%) round (75%)

In Table 4.3, each bold adjectives shows the correctly predicted category. The first column includes the images of the testing objects, the second and third columns show the accuracies with and without co-occurrence information, relatively, and the last column is the testing accuracy for trained SVM data. For object O_2 , SVM and only perceptual similarity classify this object as thin, but co-occurrence corrects the wrong prediction. We can also conclude that our method has better prediction accuracy than SVM if we look at the objects O_2 and O_5 .

In Table 4.4, the predicted noun categories using co-occurrence for each object conforms to the result of perception-only predicted nouns, and the predictions using with/out co-occurrence are more accurate than using SVM. More specifically, the noun category of the object O_2 is wrongly predicted by SVM, whereas it is correctly predicted from perceptual features.

Table 4.4: Predicted nouns for some objects from the test set (bold denotes correct classification). The co-occurrence weight w_{an} is taken as 0.2.

Objects	Nouns		
	Perceptual Similarity (s_{perc}^n)	Perc. Similarity and Co-occurrence s_{comb}^n	SVM
 O_1	Box (22%) Cylinder (24%) Cup (37%) Ball (17%)	Box (23%) Cylinder (24%) Cup (35%) Ball (18%)	Box (25%) Cylinder (23%) Cup (45%) Ball (7%)
 O_2	Box (32%) Cylinder (30%) Cup (19%) Ball (19%)	Box (36%) Cylinder (34%) Cup (15%) Ball (15%)	Box (38%) Cylinder (44%) Cup (3%) Ball (15%)
 O_3	Box (34%) Cylinder (25%) Cup (21%) Ball (20%)	Box (32%) Cylinder (25%) Cup (22%) Ball (21%)	Box (67%) Cylinder (16%) Cup (4%) Ball (13%)
 O_4	Box (22%) Cylinder (23%) Cup (20%) Ball (35%)	Box (22%) Cylinder (23%) Cup (22%) Ball (33%)	Box (3%) Cylinder (3%) Cup (1%) Ball (93%)
 O_5	Box (24%) Cylinder (47%) Cup (16%) Ball (13%)	Box (24%) Cylinder (43%) Cup (18%) Ball (15%)	Box (34%) Cylinder (44%) Cup (6%) Ball (16%)

It can be easily inferred that the co-occurrence information improves the prediction accuracies, given in Tables 4.3, 4.4, and Figures 4.1, 4.2. Although the perceptual similarity confidences are accurate enough for noun category prediction for our test objects, it is not possible to get reliable predictions for all kind of objects. At this point, we postulate that we can enhance the accuracies using co-occurrence infor-

mation between noun and adjective categories by indicating that this approach is not only the solution to this problem, but one of the possible way of suppressing wrong predictions.

Table 4.5: “What object is it?” game: Determine noun based on given adjectives.

Given Adjectives				Predicted Nouns
a_1	a_2	a_3	a_4	
hard	short	thick	edgy	Box (73%) Cup (53%)
hard	round	-	-	Cup (72%) Cylinder (70%)
silent	short	thick	round	Ball (70%) Cup (52%)
short	thick	-	-	Cup (69%) Ball (69%)
round	thin	soft	tall	Ball (18%) Cylinder (17%)
soft	silent	thick	-	Ball (45%) Cup (23%)
hard	noisy	round	-	Cup (73%) Cylinder (46%)
short	thick	round	-	Ball (70%) Cup (69%)
edgy	-	-	-	Box (100%) Others (0%)

4.3.3 The “What object is it?” Game

This game is devised to find the possible noun categories from a given set of adjectives either manually, or perceptually. The aim of this game is to demonstrate the impact of the interaction between noun and adjective concepts.

Table 4.5 shows the noun category with confidences of given sample sets of adjectives. Some noun category has characteristic adjectives *per se*. For instance, *cups* and *cylinders* are always *round* and *hard*, and *boxes* are inherently *edgy*. If any adjective set conforms to this property, then the resultant noun category is strongly predicted with high confidence value. Otherwise, the confidence values will remain at low levels.

4.3.4 Concept Labeling Accuracy

We have used cross-situational labeling previously mentioned in Section 4.2. In our experiment, we have only one noun category for each training object and there is exactly one noun label for each row (object) in training label set. For noun concept labeling, we get 100% accuracy. All the diagonal cells are filled up with values of this table. For adjective concept labeling process, the accuracy results show that our modified version of labeling method suits more in our dataset in Figure 4.3. There is one-to-one correspondence between concept names and the adjective names. Although there can be more than one label for an adjective concept, we can correctly select the corresponding label for adjective concepts.

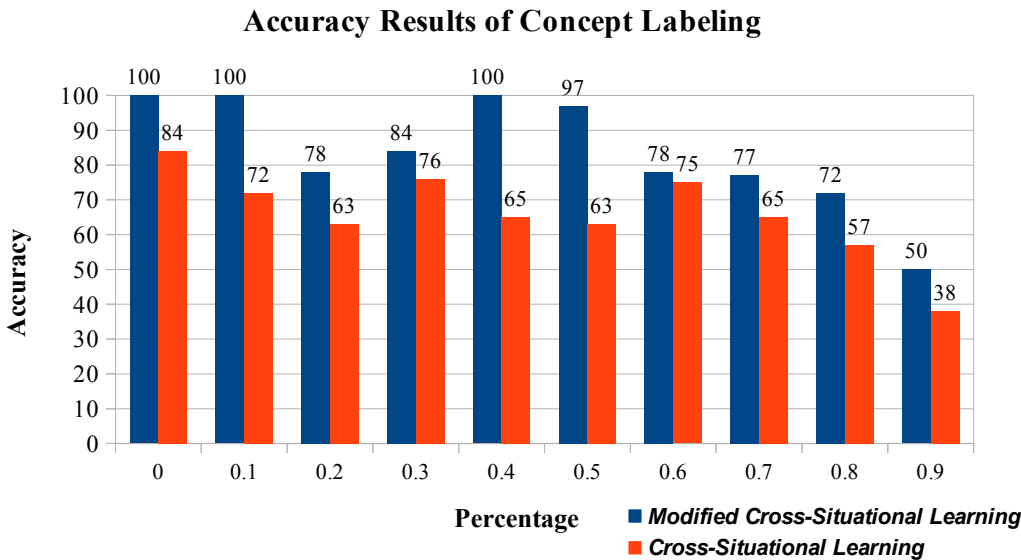


Figure 4.3: Accuracy result for concept labeling. The horizontal axis shows the percentage of included subsets of adjectives, while the vertical axis is the accuracy for correct labeling of concepts

CHAPTER 5

A WEB OF CONCEPTS

In this chapter, we present how we can build and model a web of concepts, and demonstrate its use over different test scenarios.

5.1 Building a Web of Concepts

Although co-occurrence provides us better prediction of adjective and noun categories, we want to add to the system contextual information and also (e.g., web concepts) defined by human actor. This system uses the co-occurrence information between noun, adjective, and verb categories, the context and words that are spoken by a human. This enhancement provides the system a wider knowledge of the environment, and so, better prediction of concepts. Moreover, the concept web provides the activation of concepts with respect to the connection strength between other activated concepts. The robot connects what it sees, being the entities, properties, and applied behaviors, to the previously known concepts using the prototype approach developed in previous work [7, 80]. A concept (c) can be a noun ($c \in \mathcal{N}$), adjective ($c \in \mathcal{A}$), or verb ($c \in \mathcal{V}$), with $\mathcal{C} = \mathcal{N} \cup \mathcal{A} \cup \mathcal{V}$. The activated concepts can also activate the other related concepts, where relatedness is extracted from co-occurrences in the interactions of the robot.

We model the web as a Markov Random Field (MRF) since it naturally fits as one, being composed of “nodes” that are connected to each other and that activate each other. Activation of concepts in concept web is usually performed using the message passing or belief propagation algorithms described by Koller [84] and Gouws [2].

However, our concept web, consisting of maximal cliques of concepts, is a cyclic graph, so we used another algorithm, named *Loopy Belief Propagation*, explained in Section 2.5.2.

5.1.1 Integrating LBP into Web of Concepts

The methods explained in Chapter 2 as background are all used in *acyclic* graphs. Nevertheless, our web of concepts includes a large number of nodes, including perceptual, language, verb, and effect concepts. Perceptual concepts directly come from the adjective (\mathcal{A}) and noun (\mathcal{N}) concepts. Superordinate concepts are created by a human actor, and properties that cannot be sensed directly. Some examples are *covered*, *metal*, *toy*, *etc.* This information is regarded as ground truth and transferred directly into the system. Such nodes are initialized with an activation of 100%. Verb concepts are the concepts created by the repertoire of our behaviors. These behaviors can be learned from effect features as previously mentioned. Finally, the effect concepts are the concepts that demonstrate an outcome of a behavior over an object. A label of an effect concept over an object is predicted using Support Vector Machine (SVM).

Due to this complex structure of our graph (Figure 5.1), we cannot employ standard message passing algorithms. Therefore, the LBP perfectly suits our system since it can tackle such complexities.

As explained in Chapter 2, there are two types of nodes and their potential tables. The separator nodes are the nodes that represents all type of concepts. We represent the dichotomy of our adjective concept pairs as one separator node since their potential table is one dimensional. For example, *Hard* and *Soft* concepts are placed into the same node, and their probabilities are respectively $p(Hard)$ and $p(\neg Hard)$.

Another important question is how to create clique nodes and connect them to separator ones. Initially, for each training and test object, we have a set of predefined adjective, noun and superordinate categories. These categories are determined by a human. We also have a fixed set of behaviors. However, not all behaviors can be applied to each object. In Table 5.1, *Drop*, *Shake*, *Knock Down*, and *Throw* behaviors cannot be applicable to fragile objects, namely *Cup* and *Plate* since these objects can

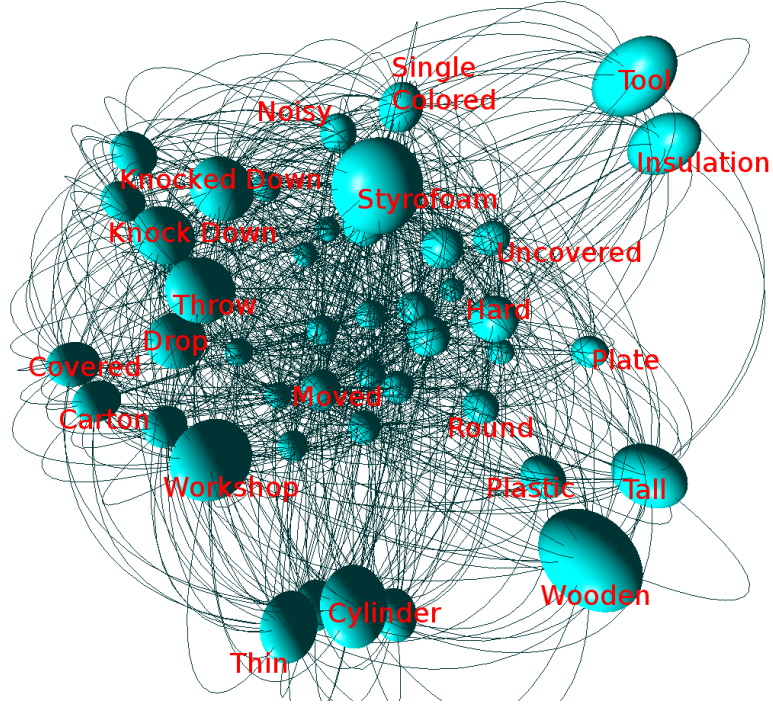


Figure 5.1: MRF representation of our web of concepts (only perceptual and language concepts included) using Ubigraph Library [5]. For the sake of comprehensibility, the labels of the background concepts are not written explicitly.

be easily broken. Thus, the applicable behaviors for each object are fed to the system. We also guarantee that there is no one-hop connection between behavior concepts, but they can be connected to each other through other noun or adjective concepts. The connections between other concepts are created in a similar manner. Finally, the connections between superordinate concepts are created with the supervision of a human actor. One important issue is that a concept which has a dichotomy with another concept is placed in the same separator node as previously mentioned. After creating the MRF graph, we find maximal cliques to convert it into a factor graph, since the LBP algorithm works on factor graphs.

After creating the nodes and connections, the potential tables of clique nodes are filled with co-occurrence information as in Chapter 4. They are multi-dimensional matrices, whose dimension size is the number of concepts placed into that clique node. Initially, the potential tables of separator nodes are filled with perceptual prediction confidences, formulated in Equations 3.4, and 3.5. The belief propagation process will then commence with the message from separator node to the clique node

Table 5.1: Possible applicable set of behaviors with respect to object categories. (*arg*: *Left, Right, Forward, Backward*; A: *Applicable*; NA: *Not-Applicable*)

	Push(<i>arg</i>)	Move(<i>arg</i>)	Drop	Grasp	Shake	Knock Down	Throw
Box	A	A	A	A	A	A	A
Ball	A	A	A	A	A	A	A
Cylinder	A	A	A	A	A	A	A
Cup	A	A	NA	A	NA	NA	NA
Tool	A	A	A	A	A	A	A
Plate	A	A	NA	A	NA	NA	NA

spreading the initial confidences around the web. As a special case, the potential tables of the effect nodes are filled using the Gibson’s notion of affordances [85]. For each testing object and behavior, we get the confidence values from SVM effect prediction and insert them as a potential value of an effect concept in a web. For example, we predict the effect confidence value of a *ball* after applying *grasp* behavior ($e_{ball}, b_{grasp}, f_{predicted}$). In other words, the effect nodes are not physically connected to other nodes. Their prediction results are obtained directly from SVM.

5.2 Results

In this section, we give two possible scenarios to demonstrate the contributions of having a web of concepts.

5.2.1 Scenario 1: “Perception-Driven Activation of Concepts in the Web”

In the first scenario, iCub encounters an unknown object, and tries to guess the noun and the adjectives of this object. In addition, it tries to foresee what kind of actions are possibly applicable on this object, together with their possible effect outcomes.

Initially, we place the object on a table in front of iCub. iCub examines the object visually, as well as grasping and shaking it to check its haptic, auditory and proprioceptive properties (Section 3.3). The entity feature vector is extracted out of these sensory data (Section 3.4). After that, using the extracted features, and comparing them to the previously obtained prototypes, it predicts the probable adjective and noun categories for the object (Section 3.6). Additionally, the human trainer has the

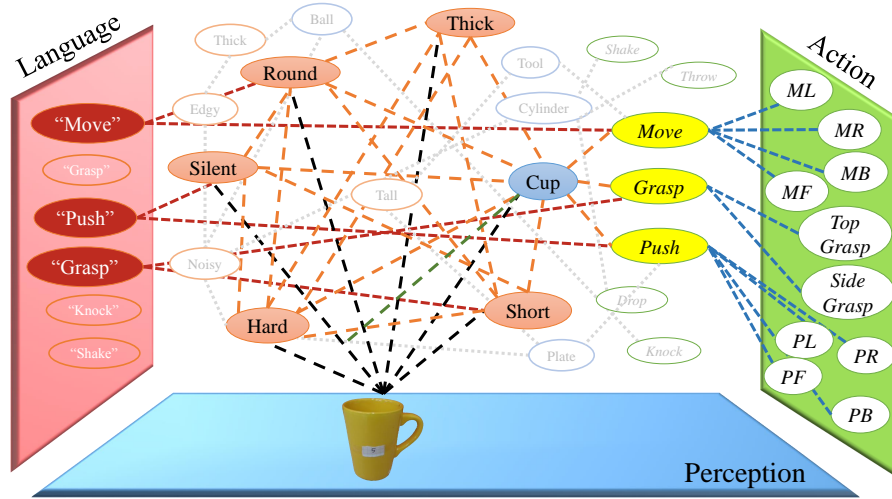


Figure 5.2: Schematized representation of Scenario 1. The *Cup* is given to the system and all related concepts are activated. The action space and verb concepts are contoured with green, whereas blue and orange colors represent the noun and adjective categories for the object, respectively. The gray and smaller fonts show inactive concepts in the web, while bigger fonts and colored nodes represent activated concepts. There are other concepts that are not shown for clarity. (*ML*: Move Left, *MR*: Move Right, *MF*: Move Forward, *MB*: Move Backward, *PL*: Push Left, *PR*: Push Right, *PF*: Push Forward, *PB*: Push Backward)

option of specifying extra knowledge about the object, such as its material (*wooden, plastic, metal, etc.*) or function (*utencil, workshop, toy*), that iCub cannot detect itself with its limited sensory data. We postulate that, if available, these new concepts will also enhance our prediction accuracies.

When all the information that can be collected is gathered, this information is fed into the concept web, whose connections has been previously determined using the statistical properties of the training data (Section 5.1.1). The concepts that are predicted by iCub and the knowledge nodes provided by the human are initially activated. All other unknown nodes are initialized with a probability of 0.5. Then the concept web is allowed to propagate activation until convergence. When convergence is established, three things have happened: (1) iCub has refined its a priori guesses about the noun and adjective categories of the object, possibly correcting some wrong guesses. (2) iCub has predicted which behaviors are applicable to this object, purely due to the

connectivity properties of the concept web. (3) iCub furthermore predicts the possible effects of these behaviors on the object, using the trained SVMs with the extracted feature vector of the object (Section 3.6.4).

An example scenario is shown in Figure 5.2. In this case, a cup is presented to iCub, and it is expected to predict as much as it can about this object. As can be seen in the figure, iCub correctly determines that the given object is a cup, and it is also hard, round, thick, short, and silent. Furthermore, it decides that the object can be grasped (after which the object will be *lifted*), moved and pushed (after which the object will be *moved*).

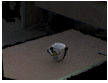





Another example might be the presentation of a ball. This time all the behaviors are applicable to this object, therefore all verb concepts (\mathcal{V}) are activated. In addition, the related noun and adjective concepts are activated which are *Ball*, *Round*, *Soft*, *Short*, *Silent*, *Thick*.

We propose that this behavior of the system is similar to the characteristics of canonical neurons [86, 87, 88]. The canonical neurons are visuomotor neurons: They respond selectively to certain behaviors, however they also respond when the subject sees an object on which this specific behavior can be applied. For instance grasping canonical neurons fire (1) when grasping, (2) when a graspable object is seen. The output of our system is also similar, i.e, verb concepts activate to objects affording the corresponding behaviors.

We now apply this scenario to predict the categories of an object. We present 6 sample objects to iCub, each one selected from a different noun category. Then we make him predict the adjective and noun categories of objects, as well as the applicable behaviors and their effects for each object. To show the effectiveness of this approach, the predicted categories using only perceptual similarities, explained in Section 3.6 are compared to using the web of concept.

The results are depicted in Table 5.2. The first column shows the RGB-colored depth images of the instance for each noun category, that are taken from Kinect sensor using PCL modules. The second and third columns show the perception-only predictions for noun and adjective categories, respectively. The fourth and fifth columns show

Table 5.2: The prediction accuracies of noun and adjective categories using the concept web, with respect to the perception-only guesses. 6 objects, one of each noun category, are used for demonstration. Images depict RGB-colored depth images (collected via PCL library from the Kinect sensor). The second and third columns depict the perception-only results, while the fourth and fifth columns show the concept web predictions. Prediction confidences are indicated in paratheses. The use of bold text indicates correct decisions. Striked-out text indicates wrong decisions. [Best viewed in color]

Object	Predicted Nouns (Perception only) (% confidence)	Predicted Adjectives (Perception only) (% confidence)		Predicted Nouns (Concept web) (% confidence)	Predicted Adjectives (Concept web) (% 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 (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 (11%) box (14%) cup (17%) cylinder (31%) plate (10%) tool (17%)	edgy (40%) hard (64%) noisy (63%) short (44%) thick (40%)	round (60%) soft (36%) silent(37%) tall (56%) thin (60%)	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 (42%) cup (11%) cylinder (12%) plate (10%) 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 (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 (15%) box (18%) cup (16%) cylinder (17%) plate (21%) tool (13%)	edgy (44%) hard (51%) noisy (44%) short (52%) thick (53%)	round (56%) soft (49%) silent(56%) tall (47%) thin (47%)	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 %)

the prediction results obtained using concept-web counterparts.

There are two crucial contributions that the web of concepts provide to our predictions: (i) The difference between correct and wrong predictions for noun and adjective categories become more straightforward when compared perception-only results, and (ii) the wrongly predicted adjective categories that are struck out for 1st, 3rd, and 6th

objects are corrected.

Table 5.3: The predictions of applicable behaviors and their likely effects via the concept web. *NA* stands for *Not-Applicable*. The confidence values of the predictions are unanimously (100%), so are intentionally not shown for clarity. Images depict RGB-colored depth images (collected via PCL library from the Kinect sensor). [Best viewed in color]

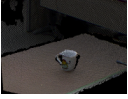





Object	Predicted Behaviors	Predicted Effects	Object	Predicted Behaviors	Predicted Effects
	grasp	lifted		grasp	lifted
	push	moved		push	moved
	move	moved		move	moved
	throw	NA		throw	moved
	drop	NA		drop	moved
	knock down	NA		knock down	moved
	shake	NA		shake	moved
	grasp	lifted		grasp	lifted
	push	disappeared		push	moved
	move	moved		move	moved
	throw	disappeared		throw	moved
	drop	disappeared		drop	moved
	knock down	moved		knock down	knocked
	shake	disappeared		shake	moved
	grasp	lifted		grasp	lifted
	push	moved		push	moved
	move	moved		move	moved
	throw	moved		throw	NA
	drop	moved		drop	NA
	knock down	knocked		knock down	NA
	shake	moved		shake	NA

Table 5.5 depicts the predictions of applicable behaviors and their likely effects on the same objects. The predicted results are all correct, with cup and plate objects detected as being unable to be thrown, dropped, shaken, and knocked down, and ball objects rolling down the table when pushed. The prediction accuracies are 100% in each trial, and therefore has not been stated individually for clarity.

5.2.2 Scenario 2: “Interaction-Driven Activation of Concepts in the Web”

In the second scenario (Figure 5.3), human actor gives an unknown object to iCub, and commands it to apply a specific behavior on this object. The activation in this case progresses from two different channels. In this first pathway, iCub looks at the object and determines its visual properties. It uses these visual properties to extract a partial feature vector (excluding haptic, proprioceptive and auditory features). In the second path, a human trainer commands iCub to apply a behavior to the object. This specified behavior directly activates the related verb concept in the web. As

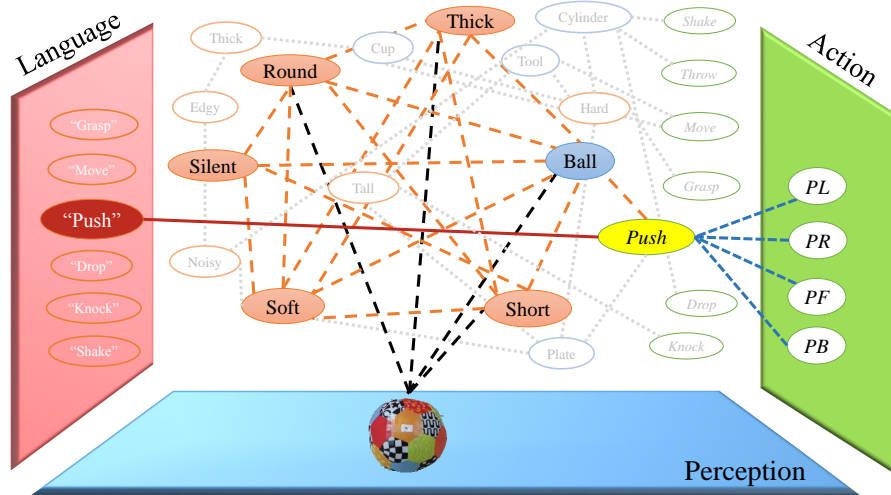


Figure 5.3: Schematized representation of Scenario 2. The *Ball* is given to the system and *Push* behavior is commanded to iCub. All related concepts are activated (only visually perceivable concepts). The action space and verb concepts are contoured with green, whereas blue and orange colors represent the noun and adjective categories for the object, respectively. The gray and smaller fonts show inactive concepts in the web, while bigger fonts and colored nodes represent activated concepts. There are other concepts that are not shown for clarity. (*PL*: Push Left, *PR*: Push Right, *PF*: Push Forward, *PB*: Push Backward)





in Scenario 1, all of the clique potentials are filled using co-occurrence information (Section 5.1.1). The confidence values of the visually found concepts of the given object are also placed into the corresponding separator nodes. All of the probabilities for other concepts are left balanced (0.5).

In this scenario shown in Figure 5.3, a ball is put on the table, and “push” behavior is requested. Although they are not predicted perceptually (due to the initially missing haptic, proprioceptive and auditory features), the *Soft* and *Silent* concepts are correctly found. Moreover, after the activation of verb concept *push* with the command of a human, all the related actions, namely *Push-Left*, *Push-Right*, *Push-Forward*, and *Push-Backward*, are activated. Finally, the most probable effect outcome is estimated using the associated SVM of the grasp behavior on the collected visual features. Since the balls roll down from the table and disappear when pushed, the predicted effect is “Disappeared”.

In another example, *Cylinder* object is served to iCub, and we want to perform “grasp” behavior on the object. Initially, all visually perceivable concepts which are *Tall*, *Thin*, *Round*, *Cylinder* are activated with *Grasp* verb concept. Grasp concept also activates *Top-Grasp* and *Side-Grasp* concepts in action space. After convergence, other related concepts (*Noisy*, *Hard*) also become activated.

The aim of this scenario is to show that it is possible to find all concepts of the pre-served object using only restricted set of features (in this case, visual features) even if any behavior is not performed on this object. Also it shows how an issued action command activation spreads in parallel through the web.

Table 5.4: The prediction accuracies of noun and adjective categories on the *novel* objects using the concept web. Initially, only visually activated concepts are perceptually predicted, and the activations are spread to predict all related concepts. 4 novel objects, are used for demonstration. The second and third columns depict the perception-only results, while the fourth and fifth columns show the concept web predictions. Prediction confidences are indicated in paratheses. The use of bold text indicates correct decisions. Striked-out text indicates wrong decisions. [Best viewed in color]

Objects	Predicted Nouns (Perception Only) (% confidence)	Predicted Adjectives (Perception Only) (% confidence)	Predicted Nouns (Concept Web) (% confidence)	Predicted Adjectives (Concept Web) (% confidence)
	ball (11%) box (16%) cup (14%) cylinder (38%) plate (12%) tool (9%)	edgy (40%) short (36%) thick (45%) round (60%) tall (64%) thin (55%)	ball (0%) box (0%) cup (0%) cylinder (100%) plate (0%) tool (0%)	edgy (0%) hard (100%) noisy (0%) short (0%) thick (0%) round (100%) soft (0%) silent (100%) tall (100%) thin (100%)
	ball (11%) box (12%) cup (13%) cylinder (26%) plate (11%) tool (27%)	edgy (37%) short (23%) thick (49%) round (63%) tall (77%) thin (51%)	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 (16%) box (23%) cup (18%) cylinder (17%) plate (14%) tool (12%)	edgy (43%) short (59%) thick (56%) tall (41%) thin (44%) round (57%)	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 (16%) box (15%) cup (16%) cylinder (20%) plate (14%) tool (19%)	edgy (51%) short (52%) thick (54%) round (49%) tall (48%) thin (46%)	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%)

Although the concept web works well on the testing set for this scenario, we have also used novel objects which are really different when compared with test objects.

In Table 5.4, we have four novel objects. For the second and the third objects, we have wrongly predicted the noun categories and some of the adjective categories, whereas the noun category of the last object is correctly predicted. As a result of the activation of concepts, noun categories of the whole objects are wrongly predicted. The reason of this wrong noun category prediction arises from the wrongly activated adjective concepts. For the second and third objects, the activated adjective concepts are “round”, “hard”, “silent”, “tall” and “thick”, directly causing the wrong activation of “plate” concept, which is completely natural since activated concepts include the dominant adjectives of the objects which are classified as “plate” in the training set.

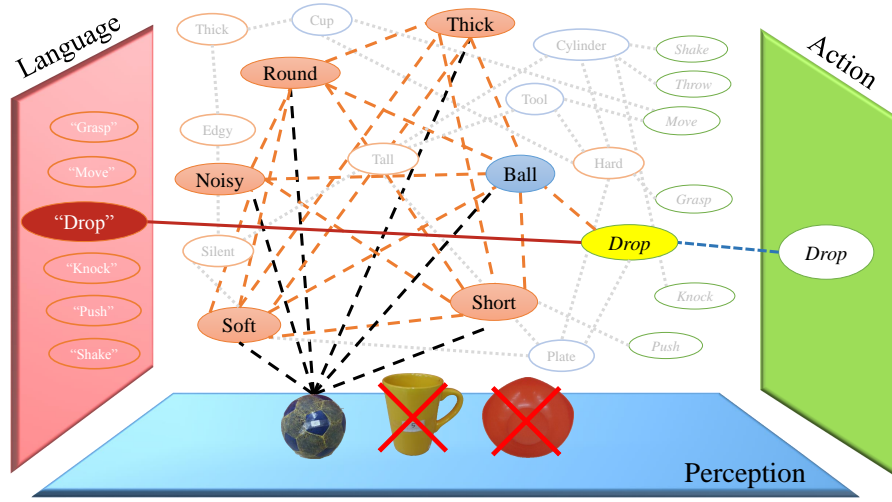




Figure 5.4: Schematized representation of Scenario 3. The sample *Ball*, *Cup*, and *Plate* objects are given to the system and *Drop* behavior is commanded to iCub. iCub selects any one of these objects if the commanded behavior is applicable. In this scenario, the *Ball* object is selected and its activated concepts are shown. The action space and verb concepts are contoured with green, whereas blue and orange colors represent the noun and adjective categories for the object, respectively. The gray and smaller fonts show inactive concepts in the web, while bigger fonts and colored nodes represent activated concepts. There are other concepts that are not shown for clarity.

5.2.3 Scenario 3: “Command-Driven Activation of Concepts in the Web”

In this final scenario, we show how iCub responds when commanded to perform a certain action in an environment populated with multiple objects (Figure 5.4). The command does not specify on which object to apply the behavior, therefore iCub must choose an appropriate object. Remember that some behaviors cannot be applied to certain types of objects. Therefore, we expect that activation will not spread from these verb concepts to inappropriate noun types. Properly activated nouns will be options for iCub to apply the behavior. If there are more than one appropriate objects, iCub makes a random decision. (Probability of being chosen for a certain noun type will be proportional to its frequency in the training set.)

In the sample scenario, iCub is presented with a cup, a plate, and a ball, and is commanded to apply “Drop” behavior. Due to the missing connections between Drop verb, and Cup and Plate nouns, activation cannot spread to Cup and Plate. On the other hand, Drop and Ball are connected, through which Ball noun is activated. As a result, iCub decides to apply this action to the ball object.

Table 5.5: The selection of objects on which sample commands are applicable behaviors. The confidence values of the predictions are unanimously (100%), so are intentionally not shown for clarity. Images depict RGB-colored depth images (collected via PCL library from the Kinect sensor). [Best viewed in color]

Command	Viewed Scene	Selected Objects
throw		box
push		box green cup white cup yellow plate red plate ball

The aim of this scenario is to show that behaviors can activate related noun concepts, while avoiding activation in the unrelated ones. This kind of “reverse” activation spreading can guide the robot’s actions in the world.

CHAPTER 6

DISCUSSION AND CONCLUSION

In this thesis, we have addressed the issue of finding a shared representation of concepts between a human actor and a humanoid robot. As we have discussed in Chapter 1 and 2, this is crucial for seamless communication with a robot since the words used by the human and the robot should activate the same meaning in their “brains”. More specifically, we have tackled the problem of learning and representing nouns, adjectives, verbs in a single model, unlike the existing studies that study these categories separately.

We first showed that, as a proof of concept, co-occurrence between nouns and adjectives can be used to better predict them. This allowed us to predict, e.g., the noun of an object from its adjectives vice versa. This was crucial since existing studies have learned nouns and adjectives separately. However, from studies in Neuroscience, we know that humans represent and activate concepts not only based on their perceptual or sensorimotor information but also by using the other concepts in our brain.

We then tackled the more general problem of modeling a web of concepts in a robot that include not only nouns and adjectives but also verbs, effects, language, higher-level noun and adjective categories, and of course the links between them. This is an important attempt in the literature for modeling a web of concepts that gets activated in a fashion similar to humans. We modeled the web as a Markov Random Field and made inferences using Loopy Belief Propagation as they proved to be very suitable for such complex graphs. We showed that this web allowed the robot to activate the relevant concepts of an object by just looking at the object. With this activation, for example, the robot knows the noun, the adjectives, the words that can be used for the

object as well as the applicable behaviors.

6.1 Limitations and Future Work

We have a limited set of objects for our experiments. Increasing the number of objects in the object set affects our system with respect to reliability. As the number of objects increases, the prototypes might capture more detailed representation of concepts. Moreover, the co-occurrence values are enhanced, since the ambiguity due to the limited object set can be corrected with increasing number of exemplars of a category. For example, one contradicting example in a category with five objects deteriorates more than in a category with 10 objects. Therefore, the more exemplars there are in our training set, the more reliable our system is. On the other hand, increasing the number of objects does not affect the computational complexity if the cardinality of the adjective and noun categories remains the same. The computational complexity directly depends on the number of concepts and their connections.

The concepts in the web have an ontological structure. There are other studies using ontological structure to represent the concepts [89, 90, 91]. For this kind of systems, there is a relation between concepts, and the relation is expressed using probabilities. Integration of an ontological system into our system can easily be done by arranging the relation criteria and adding non-existing connections and concepts into the web.

Our model lacks on-line formation of concept web. In this thesis, the nodes and connections are directly created over the entire set of training objects. We have not developed any incremental web formation method. For newly added objects, we reconstruct the web, which is not a realistic for human learning since the concept learning is an incremental process that going on throughout entire human life. This can be done by adding object categories to the existing web by testing whether it is newly introduced or not, and updating all co-occurrence information of the resultant web.

Another important improvement can be the modeling of long and short-term memory, as depicted in Figure 6.1. The long-term memory can be thought as a combination of “situated concepts” that are associated with the contextual information, gathered

from previously experiences, while the short-term memory is the representation of the current scene. There can be connections between long- and short-term memories to check whether a desired concept is any one of them or not. The concepts in short-term memory is instantaneous while the latter one holds general information.

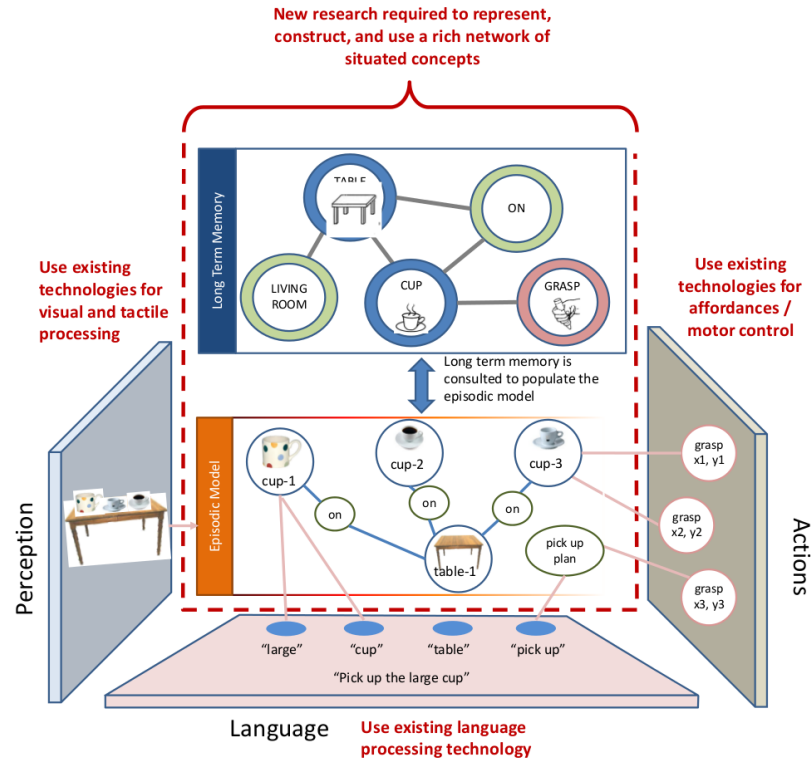


Figure 6.1: Representation of Long and Short Term Memories consisting of “situated” concepts and instantaneous concepts related with behavior and object, respectively.

Finally, the set of concepts can be enriched by adding more concepts to the system. This can be done by adding a completely new concept that are obtained from different modalities or an intermediate concept for existing concepts. For instance, “medium-length” concept can be added between “tall”, and “short” adjective concepts.

REFERENCES

- [1] John H. Gennari, Pat Langley, and Doug Fisher. Models of incremental concept formation. *Artificial Intelligence*, 40(1–3):11 – 61, 1989.
- [2] Almero Gouws. *A Python implementation of graphical models*. PhD thesis, Stellenbosch: University of Stellenbosch, 2010.
- [3] Support vector machine. http://en.wikipedia.org/wiki/Support_vector_machine, 2014. Last visited date: August 4, 2014.
- [4] Hande Çelikkanat, Güner Orhan, Nicolas Pugeault, Frank Guerin, Erol Şahin, and Sinan Kalkan. Learning and using context on a humanoid robot using latent dirichlet allocation. In *Development and Learning and Epigenetic Robotics (ICDL), 2014 IEEE Third Joint International Conference on*, pages 1–6, Aug 2014. Accepted.
- [5] Ubigraph: Free dynamic graph visualization software. Last visited date: July 24, 2014.
- [6] Douglas H. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2(2):139–172, 1987.
- [7] Sinan Kalkan, Nilgün Dag, Onur Yürüten, Anna M. Borghi, and Erol Şahin. Verb concepts from affordances. *Interaction Studies*, 15:1–37(36), 2014.
- [8] Jivko Sinapov, Connor Schenck, Kerrick Staley, Vladimir Sukhoy, and Alexander Stoytchev. Grounding semantic categories in behavioral interactions: Experiments with 100 objects. *Robotics and Autonomous Systems*, 62(5):632 – 645, 2014. Special Issue Semantic Perception, Mapping and Exploration.
- [9] Onur Yürüten, Kadir F. Uyanık, Yiğit Çalışkan, Asil Kaan Bozcuoğlu, Erol Şahin, and Sinan Kalkan. Development of adjective and noun concepts from affordances on the icub humanoid robot. *12th International Conference on Adaptive Behaviour (SAB)*, 2012.
- [10] Angelo Cangelosi. Grounding language in action and perception: From cognitive agents to humanoid robots. *Physics of Life Reviews*, 7(2):139 – 151, 2010.
- [11] Terrence Deacon. The symbolic species: the co-evolution of language and the human brain, 1997.

- [12] Joanna J. Bryson. Embodiment versus memetics. *Mind & Society*, 7(1):77–94, 2008.
- [13] Tom M. Mitchell, Svetlana V. Shinkareva, Andrew Carlson, Kai-Min Chang, Vicente L. Malave, Robert A. Mason, and Marcel Adam Just. Predicting human brain activity associated with the meanings of nouns. *Science*, 320(5880):1191–1195, 2008.
- [14] John L. Locke. Babbling and early speech: Continuity and individual differences. *First Language*, 9(6):191–205, 1989.
- [15] Friedemann Pulvermüller. *The neuroscience of language: on brain circuits of words and serial order*. Cambridge University Press, 2002.
- [16] Fei Xu. The role of language in acquiring object kind concepts in infancy. *Cognition*, 85(3):223 – 250, 2002.
- [17] Kim Plunkett, Jon-Fan Hu, and Leslie B. Cohen. Labels can override perceptual categories in early infancy. *Cognition*, 106(2):665 – 681, 2008.
- [18] Stevan Harnad. The symbol grounding problem. *Physica, D*, 42:335–346, 1990.
- [19] Angelo Cangelosi and Domenico Parisi. The processing of verbs and nouns in neural networks: Insights from synthetic brain imaging. *Brain and Language*, 89(2):401 – 408, 2004. Language and MotorIntegration.
- [20] Luc Steels, Frederique Kaplan, Angus McIntyre, and Joris Van Looveren. Crucial factors in the origins of word-meaning. *The transition to language*, 12:252–271, 2002.
- [21] Luc Steels. Evolving grounded communication for robots. *Trends in Cognitive Sciences*, 7(7):308 – 312, 2003.
- [22] Linda Smith and Chen Yu. Infants rapidly learn word-referent mappings via cross-situational statistics. *Cognition*, 106(3):1558 – 1568, 2008.
- [23] Chen Yu and Linda B. Smith. Rapid word learning under uncertainty via cross-situational statistics. *Psychological Science*, 18(5):414–420, 2007.
- [24] Anthony F. Morse, Paul Baxter, Tony Belpaeme, Linda B. Smith, and Angelo Cangelosi. The power of words. In *Joint IEEE International Conference on Development and Learning and on Epigenetic Robotics*, 2011.
- [25] Liane Gabora, Eleanor Rosch, and Diederik Aerts. Toward an ecological theory of concepts. *Ecological Psychology*, 20(1):84–116, 2008.
- [26] Jerome Seymour Bruner and George Allen Austin. *A study of thinking*. Transaction Publishers, 1986.

- [27] Douglas L. Medin and Edward E. Smith. Concepts and concept formation. *Annual Review of Psychology*, 35(1):113–138, 1984.
- [28] Eleanor H. Rosch. Natural categories. *Cognitive Psychology*, 4(3):328 – 350, 1973.
- [29] Robert M. Nosofsky, John K. Kruschke, and Stephen C. Mckinley. Combining exemplar-based category representations and connectionist learning rules. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 18:211–233, 1992.
- [30] Yves Rosseel. Mixture models of categorization. *Journal of Mathematical Psychology*, 46(2):178 – 210, 2002.
- [31] Peter Gärdenfors. *Conceptual spaces: The geometry of thought*. MIT press, 2004.
- [32] George Lakoff. *Women, fire, and dangerous things: What categories reveal about the mind*. Cambridge Univ Press, 1990.
- [33] John R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.
- [34] Edward A. Feigenbaum. The simulation of verbal learning behavior. In *Papers Presented at the May 9-11, 1961, Western Joint IRE-AIEE-ACM Computer Conference*, IRE-AIEE-ACM '61 (Western), pages 121–132. ACM, 1961.
- [35] Edward A. Feigenbaum and Herbert A. Simon. Epam-like models of recognition and learning*. *Cognitive Science*, 8(4):305–336, 1984.
- [36] Michael Lebowitz. Experiments with incremental concept formation: Unimem. *Machine Learning*, 2(2):103–138, 1987.
- [37] Friedemann Pulvermüller. Brain mechanisms linking language and action. *Nature Reviews Neuroscience*, 6(7):576–582, 2005.
- [38] Robert F. Goldberg, Charles A. Perfetti, and Walter Schneider. Perceptual knowledge retrieval activates sensory brain regions. *The Journal of Neuroscience*, 26(18):4917–4921, 2006.
- [39] Marion L. Kellenbach, Matthew Brett, and Karalyn Patterson. Large, colorful, or noisy? attribute-and modality-specific activations during retrieval of perceptual attribute knowledge. *Cognitive, Affective, & Behavioral Neuroscience*, 1(3):207–221, 2001.
- [40] Antonio R. Damasio. Time-locked multiregional retroactivation: A systems-level proposal for the neural substrates of recall and recognition. *Cognition*, 33(1):25–62, 1989.

- [41] Gertrude H. Eggert. *Wernicke's works on aphasia: A sourcebook and review*, volume 1. Mouton The Hague, 1977.
- [42] Matthew A. Lambon Ralph. Neurocognitive insights on conceptual knowledge and its breakdown. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1634):20120392, 2014.
- [43] Linda L. Chao and Alex Martin. Representation of manipulable man-made objects in the dorsal stream. *Neuroimage*, 12(4):478–484, 2000.
- [44] Karalyn Patterson, Peter J Nestor, and Timothy T Rogers. Where do you know what you know? the representation of semantic knowledge in the human brain. *Nature Reviews Neuroscience*, 8(12):976–987, 2007.
- [45] Alex Martin. The representation of object concepts in the brain. *Annu. Rev. Psychol.*, 58:25–45, 2007.
- [46] Matthew A. Lambon Ralph, Karen Sage, Roy W. Jones, and Emily J. Mayberry. Coherent concepts are computed in the anterior temporal lobes. *Proceedings of the National Academy of Sciences*, 107(6):2717–2722, 2010.
- [47] Hanna Damasio, Daniel Tranel, Thomas Grabowski, Ralph Adolphs, and Antonio Damasio. Neural systems behind word and concept retrieval. *Cognition*, 92(1):179–229, 2004.
- [48] Catherine J. Mummery, Karalyn Patterson, CJ Price, J. Ashburner, RSJ Frackowiak, John R. Hodges, et al. A voxel-based morphometry study of semantic dementia: relationship between temporal lobe atrophy and semantic memory. *Annals of neurology*, 47(1):36–45, 2000.
- [49] Holly Robson, Roland Zahn, James L. Keidel, Richard J. Binney, Karen Sage, and Matthew A. Lambon Ralph. The anterior temporal lobes support residual comprehension in wernicke's aphasia. *Brain*, 137(3):931–943, 2014.
- [50] W. Simmons and Alex Martin. The anterior temporal lobes and the functional architecture of semantic memory. *Journal of the International Neuropsychological Society*, 15(05):645–649, 2009.
- [51] F. Gregory Ashby and W. Todd Maddox. Human category learning. *Annual Review of Psychology*, 56(1):149–178, 2005. PMID: 15709932.
- [52] Chen Yu and Dana H. Ballard. On the integration of grounding language and learning objects. In *AAAI*, volume 4, pages 488–493, 2004.
- [53] Eleanor J. Gibson. Exploratory behavior in the development of perceiving, acting, and the acquiring of knowledge. *Annual review of psychology*, 39(1):1–42, 1988.

- [54] V. Chu, I. McMahon, L. Riano, C.G. McDonald, Qin He, J. Martinez Perez-Tejada, M. Arrigo, N. Fitter, J.C. Nappo, T. Darrell, and K.J. Kuchenbecker. Using robotic exploratory procedures to learn the meaning of haptic adjectives. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 3048–3055, May 2013.
- [55] Allison Petrosino and Kevin Gold. Toward fast mapping for robot adjective learning. 2010.
- [56] H. Dindo and D. Zambuto. A probabilistic approach to learning a visually grounded language model through human-robot interaction. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 790–796, 2010.
- [57] Antonio Chella, Haris Dindo, and Daniele Zambuto. Grounded human-robot interaction. In *AAAI Fall Symposium: Biologically Inspired Cognitive Architectures*, 2009.
- [58] Arthur M. Glenberg and Vittorio Gallese. Action-based language: A theory of language acquisition, comprehension, and production. *Cortex*, 48(7):905 – 922, 2012.
- [59] Kevin Gold, Marek Doniec, Christopher Crick, and Brian Scassellati. Robotic vocabulary building using extension inference and implicit contrast. *Artificial Intelligence*, 173(1):145 – 166, 2009.
- [60] Pascal Haazebroek, Saskia van Dantzig, and Bernhard Hommel. A computational model of perception and action for cognitive robotics. *Cognitive Processing*, 12(4):355–365, 2011.
- [61] Sugita Yuuya, Jun Tani, and Butz Martin V. Simultaneously emerging braitenberg codes and compositionality. *Adaptive Behavior*, 19(5):295–316, 2011.
- [62] Aneesh Chauhan and Luís Seabra Lopes. Using spoken words to guide open-ended category formation. *Cognitive Processing*, 12(4):341–354, 2011.
- [63] Shane Griffith, Jivko Sinapov, M. Miller, and Alex Stoytchev. Toward interactive learning of object categories by a robot: A case study with container and non-container objects. In *Development and Learning, 2009. ICDL 2009. IEEE 8th International Conference on*, pages 1–6, June 2009.
- [64] Anna Gorbenko and Vladimir Popov. Self-learning algorithm for visual recognition and object categorization for autonomous mobile robots. In Xingui He, Ertian Hua, Yun Lin, and Xiaozhu Liu, editors, *Computer, Informatics, Cybernetics and Applications*, volume 107 of *Lecture Notes in Electrical Engineering*, pages 1289–1295. Springer Netherlands, 2012.

- [65] Christopher M. Bishop. *Pattern Recognition and Machine Learning*, volume 1. Springer New York, 2006.
- [66] Ross Kindermann, James Laurie Snell, et al. *Markov random fields and their applications*, volume 1. American Mathematical Society Providence, RI, 1980.
- [67] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.
- [68] Vladimir Naumovich Vapnik and Vladimir Vapnik. *Statistical learning theory*, volume 2. Wiley New York, 1998.
- [69] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.
- [70] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, 2009.
- [71] Giorgio Metta, Giulio Sandini, David Vernon, Lorenzo Natale, and Francesco Nori. 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, 2008.
- [72] Paul Fitzpatrick, Giorgio Metta, and Lorenzo Natale. Towards long-lived robot genes. *Robotics and Autonomous systems*, 56(1):29–45, 2008.
- [73] Radu B. Rusu and S. Cousins. 3d is here: Point cloud library (pcl). In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1–4, May 2011.
- [74] Gaël Guennebaud, Benoît Jacob, et al. Eigen v3. <http://eigen.tuxfamily.org>, 2010. Last visited date: July 24, 2014.
- [75] Jan J. Koenderink and Andrea J. van Doorn. Surface shape and curvature scales. *Image and Vision Computing*, 10(8):557 – 564, 1992.
- [76] A.K. Qin and P.N. Suganthan. Robust growing neural gas algorithm with application in cluster analysis. *Neural Networks*, 17(8-9):1135 – 1148, 2004.
- [77] Tin Lay Nwe, Say Wei Foo, and Liyanage C. De Silva. Speech emotion recognition using hidden markov models. *Speech Communication*, 41(4):603 – 623, 2003.
- [78] Eleanor Rosch. Reclaiming concepts. *Journal of Consciousness Studies*, 6(11-12):61–77.

- [79] Chen Yu and Linda B. Smith. Rapid word learning under uncertainty via cross-situational statistics. *Psychological Science*, 18(5):414–420, 2007.
- [80] Güner Orhan, Sertaç Olgunsoylu, Erol Şahin, and Sinan Kalkan. Co-learning nouns and adjectives. In *Development and Learning and Epigenetic Robotics (ICDL), 2013 IEEE Third Joint International Conference on*, pages 1–6, Aug 2013.
- [81] Michael Gasser and Linda B. Smith. Learning nouns and adjectives: A connectionist account. In *Language and Cognitive Processes*, pages 269–306, 1998.
- [82] Catherine Sandhofer and Linda B. Smith. Learning adjectives in the real world: How learning nouns impedes learning adjectives. *Language Learning and Development*, 3(3):233–267, 2007.
- [83] Galit W. Sassoon. Adjectival vs. nominal categorization processes: The rule vs. similarity hypothesis. *Belgian Journal of Linguistics*, 25(1):104–147, 2011.
- [84] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.
- [85] James J. Gibson. The theory of affordances. In *Perceiving, acting, and knowing: toward an ecological psychology*, pages pp.67–82. Hillsdale, N.J. : Lawrence Erlbaum Associates, 1977.
- [86] Akira Murata, Luciano Fadiga, Leonardo Fogassi, Vittorio Gallese, Vassilis Raos, and Giacomo Rizzolatti. Object representation in the ventral premotor cortex (area f5) of the monkey. *Journal of neurophysiology*, 78(4):2226–2230, 1997.
- [87] Giacomo Rizzolatti and Luciano Fadiga. Grasping objects and grasping action meanings: the dual role of monkey rostroventral premotor cortex (area f5). *Sensory guidance of movement*, 218:81–103, 1998.
- [88] Luciano Fadiga, Leonardo Fogassi, Vittorio Gallese, and Giacomo Rizzolatti. Visuomotor neurons: Ambiguity of the discharge or ‘motor’ perception? *International journal of psychophysiology*, 35(2):165–177, 2000.
- [89] Moritz Tenorth and Michael Beetz. Knowrob: A knowledge processing infrastructure for cognition-enabled robots. *The International Journal of Robotics Research*, 32(5):566–590, 2013.
- [90] Gi Hyun Lim, Il Hong Suh, and Hyowon Suh. Ontology-based unified robot knowledge for service robots in indoor environments. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 41(3):492–509, May 2011.
- [91] Douglas B. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):33–38, November 1995.

- [92] Janet L. Kolodner. Maintaining organization in a dynamic long-term memory*. *Cognitive Science*, 7(4):243–280, 1983.

APPENDIX A

HIERARCHICAL CONCEPT FORMATION METHODS

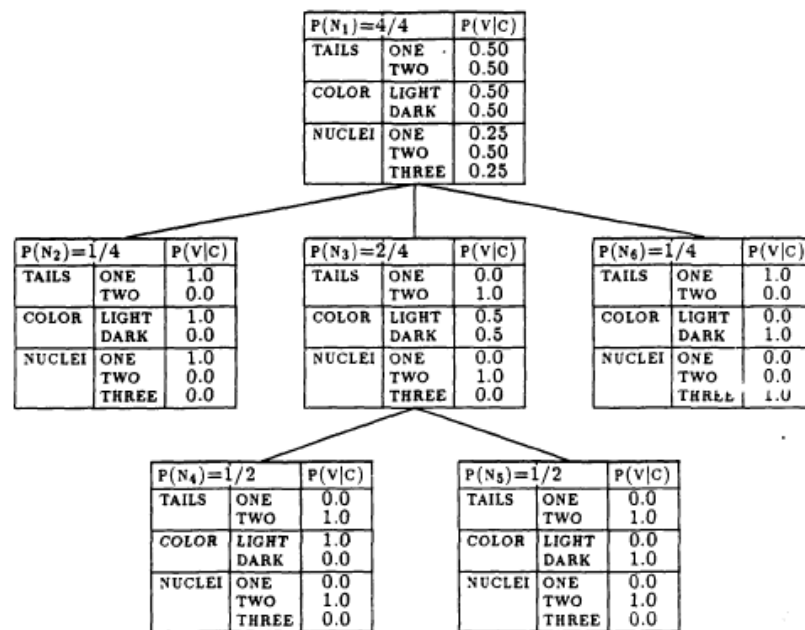


Figure A.1: COBWEB structure of each concept node. Each node is created in order of creation. Figure is taken from [1]

COBWEB [6]: The predecessors and the main inspiration sources of COBWEB are the UNIMEM and CYRUS [92]. COBWEB can be examined in four stages, which are category utility, representation of concepts, operators, and control strategy [6]. The category utility is the value of the relevance of an instance in a class and the dissimilarity of this instance to other instances in another class. In other words, it is an evaluation function that increases the intra-category similarity and inter-category dissimilarity. The representation of nodes in hierarchy is almost the same with EPAM.

It includes an attribute-value pair. As in UNIMEM, it also includes a weight for each attribute. The difference is that each node also includes a probability of occurrence of instances [1] (Figure A.1).

COBWEB has two operators to balance the concept clustering. These are *merge* and *split*. Merge operator, as it can be understood from meaning, combines two concepts by creating a parent concept of them (Figure A.2a). After placing the new instance in any concept. Split operator basically disintegrates the concept node, and adds its children nodes to the predecessor node of it (Figure A.2b). Any one of these operators is selected for each new instance by attempting them separately and testing which operator gives the best result with respect to an evaluation function.

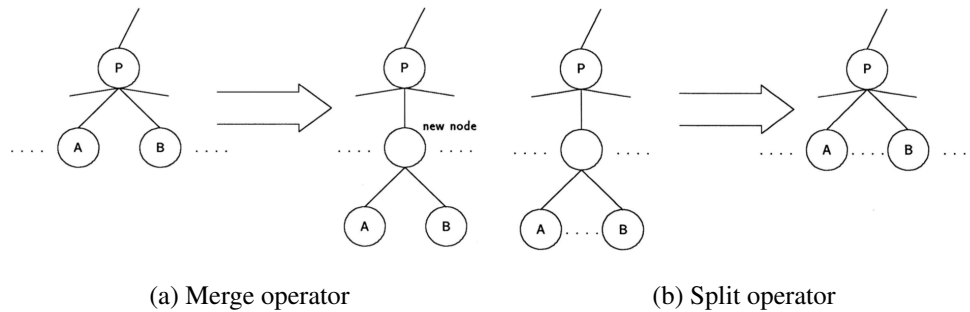


Figure A.2: The operators application procedures in COBWEB. Figure is taken from [6]

Control strategy or evaluation stage is used to determine which operator is applied with respect to category utility. It is basically determines the best representation of concept hierarchy according to instances. We can think that COBWEB is a really well-defined way of concept formation, but it has some limitations. For instance, unlike UNIMEM, it has only nominal values in nodes. As in EPAM, instances are placed into the terminal nodes.

CLASSIT [1]: This method is mostly inspired by COBWEB by keeping some necessary parts while changing the representation of concepts and instances, and evaluation function. One of the most important change is that each feature in node includes the real values, namely mean and standard deviation of the feature values. As in the previously mentioned methods, this method also keeps the more general concepts at the upper part of the tree, while more specific ones are stored in lower parts. The op-

erator selection for new instance is determined by comparing the mean and standard deviation values of a concept and a new instance. The common feature among these mentioned methods is the search methods, namely incremental hill-climbing search.

APPENDIX B

MODIFIED CROSS-SITUATIONAL LABELING

B.1 Background

In this thesis, the cross-situational labeling is used to link the concepts with our language for the sake of human-robot interaction.

Table B.1: Initial tables for adjective and noun concepts labeling

	Hard	Soft	Noisy	...
a_1	0	0	0	...
a_2	0	0	0	...
\vdots	\vdots	\vdots	\vdots	...
a_k	0	0	0	...

T_{adj}

	Box	Cylinder	Cup	Ball
n_1	0	0	0	0
n_2	0	0	0	0
n_3	0	0	0	0
n_4	0	0	0	0

T_{noun}

As in the cross-situational labeling algorithm, we get the highest valued label (column) of any one of the concepts (rows). After that, we also compare it with the highest valued concept (row) of obtained label (column). The implementation details of the modified version of the cross-situational labeling can be seen in Algorithm 2.

B.2 Algorithm

Algorithm 2: Modified Cross-Situational Based Labeling

- Create two tables for adjectives and noun concepts T_{adj}, T_{noun} (Table B.1)]

- Read one row R from the dataset. (R_i is the i^{th} element of R)

$$n_l \leftarrow R_2 \quad \mathcal{A}_l \leftarrow R_{3,4,\dots}$$

- Predict adjectives and noun concepts: \mathcal{A}_c, n_c

for each adjective concept a_c in \mathcal{A}_c and each adjective label a_l in \mathcal{A}_l **do**

$$T_{adj}^{(a_c, a_l)} \leftarrow T_{adj}^{(a_c, a_l)} + 1$$

end for

$$T_{noun}^{(n_c, n_l)} \leftarrow T_{noun}^{(n_c, n_l)} + 1$$

for each adjective concept a_c in \mathcal{A} **do**

$$L_{exc} \leftarrow \emptyset$$

while True **do**

$$p_l \leftarrow \arg \max_{w \notin L_{exc}} T_{adj}^{(a_c, w)}$$

$$L_{exc} \leftarrow L_{exc} \cup p_l$$

if $a_c = \arg \max_w T_{adj}^{(w, p_l)}$ **then**

$$LABELS(a_c) \leftarrow p_l \text{ and } \mathbf{break}$$

end if

end while

end for

for each n in \mathcal{N} **do**

$$LABELS(n) \leftarrow \arg \max_w T_{noun}^{(n, w)}$$

end for

return $LABELS$
