Politechnika Wrocławska
Wydział Informatyki i Telekomunikacji

ROZPRAWA DOKTORSKA

Kognitywna semantyka modalnych wypowiedzi o przynależności obiektu do kategorii dla przypadku systemów agentowych z wbudowaną ontologią prototypów

mgr inż. Marcin Żurawski

Promotor
dr hab. inż. Radosław Katarzyniak, prof. PWr.

Promotor pomocniczy
dr inż. Grzegorz Popek

WROCŁAW 2023

Wrocław University of Science and Technology
Faculty of Information and Communication Technology

DOCTORAL DISSERTATION

Cognitive semantics of modal class-membership statements for artificial agents with embodied prototypes-based ontology

Marcin Żurawski, M.Sc.

Supervisor
Radosław Katarzyniak, Ph.D., D.Sc.

Co-supervisor
Grzegorz Popek, Ph.D.

WROCŁAW 2023

Acknowledgements

I am extremely grateful to my supervisor Radosław Katarzyniak, Ph.D., D.Sc., for inspiration to start this project, for his time, support and encouragement during the work.

I am also thankful to my co-supervisor Grzegorz Popek, Ph.D., for his valuable insights and help, especially with the formal part of the dissertation.

I would like to extend my sincere thanks to Calisia University for funding this work.

Finally, I would like to thank my family for supporting me during the compilation of this dissertation.

Streszczenie

W rozprawie opracowano i zweryfikowano model systemu agentowego przetwarzającego wiedzę o przynależności obiektów do kategorii z prototypem i komunikującego tę wiedzę za pomocą modalnych zdań epistemicznych (w pierwszej osobie). W szczególności, rozpatrywana jest klasa zdań epistemicznych zbudowanych z modalnych operatorów wiedzy (*Wiem, że ...*; *Jestem pewien, że ...*), silnych przekonań (*Sądzę, że ...*) oraz słabych przekonań (*Możliwe, że ...*; *Uważam za możliwe, że ...*). Zdania tego typu stanowić powinny ważną grupę komunikatów językowych przetwarzanych przez interaktywne systemy agentowe.

Nadrzędnym celem badawczym zrealizowanym w rozprawie było wykazanie tezy, że możliwe jest zaprojektowanie i efektywna realizacja systemu agentowego zdolnego do przetwarzania rozpatrywanej grupy komunikatów językowych zgodnie ze zdroworozsądkowymi postulatami odzwierciedlającymi reguły przetwarzania tej klasy zdań w języku naturalnym.

Autorskim i rozszerzającym aktualny stan wiedzy osiągnięciem przedstawionym w rozprawie jest opracowanie i weryfikacja modelu przetwarzania epistemicznych zdań modalnych o przynależności obiektów do kategorii dla przypadku kategorii z prototypem. Realizacja zadania wymagała odwołania się do teorii kategoryzacji sformułowanej w obszarze lingwistyki kognitywnej, w szczególności do tzw. standardowej wersji semantyki prototypu.

Udowodnienie głównej tezy rozprawy wymagało realizacji trzech szczegółowych celów badawczych: opracowania i weryfikacji modelu wiedzy agenta, opracowania i weryfikacji modelu tzw. semantyki kognitywnej modalnych zdań epistemicznych oraz opracowania i weryfikacji przykładowych modeli implementacyjnych.

Opracowanie modelu wiedzy agenta obejmowało: zdefiniowanie wewnętrznej reprezentacji obiektów świata i oryginalnego sposobu ich przetwarzania w przestrzeni mentalnej systemu agentowego; zdefiniowanie modelu kategorii z prototypem (obejmującego jądro, granicę i obszar zewnętrzny kategorii); oraz opracowanie ogólnej strategii uczenia modelu kategorii z prototypem. Weryfikacja własności modelu wiedzy i procesu uczenia kategorii z prototypem polegała na udowodnieniu, że model kategorii uzyskany w wyniku zastosowania opracowanej strategii zachowuje właściwą relację między jądrem, granicą i obszarem zewnętrznym wyuczonej kategorii oraz na analitycznym wyznaczeniu złożoności obliczeniowej opracowanej strategii uczenia.

Opracowanie modelu semantyki kognitywnej modalnych zdań epistemicznych obejmowało: zdefiniowanie syntaktyki języka modalnej kategoryzacji wyrażającego wiedzę systemu agentowego o przynależności obiektu do kategorii z użyciem modalnych operatorów wiedzy, silnych przekonań i słabych przekonań; zdefiniowanie pojęcia otoczenia

epistemicznego; adaptację, na potrzeby semantyki kognitywnej zdefiniowanej w rozprawie, pojęcia względnej mocy gruntowania wypowiedzi modalnych wprowadzonej w teorii gruntowania modalnego języka komunikacji dla przypadku kategorii bez prototypu; szczegółowe omówienie postulowanego sposobu aplikacji semantyki kognitywnej modalnych wypowiedzi o przynależności obiektu do kategorii z prototypem; oraz opracowanie zestawu tzw. relacji epistemicznego spełnienia formuł modalnych reprezentujących poszczególne przypadki kognitywnej semantyki modalnych zdań epistemicznych o przynależności obiektu do kategorii z prototypem w wersjach dla obiektów z kompletną i niekompletną reprezentacją ich stanu. Weryfikacja oryginalnie zaproponowanego modelu semantyki kognitywnej polegała na przedstawieniu dowodów dwunastu twierdzeń o zgodności opracowanej semantyki kognitywnej z pragmatyką użycia w języku naturalnym modalnych zdań o przynależności obiektu do kategorii z prototypem oraz omówieniu związków zaproponowanej semantyki kognitywnej z teorią gruntowania modalnego języka komunikacji w systemach agentowych, sformułowaną dla przypadku kategorii bez prototypu.

W rozprawie zaproponowano i zweryfikowano trzy modele implementacyjne: model oparty na nieważonej odległości Hamminga, model oparty na ważonej odległości Hamminga i model oparty na asymetrycznej ważonej odległości Hamminga. Dla każdego z powyższych modeli opracowano dwie wersje poszukiwania zbioru kandydatów na prototyp – będących odpowiednio centroidami lub medoidami. Weryfikacja modeli implementacyjnych dla każdego przypadku obejmowała dowód twierdzenia o złożoności obliczeniowej naiwnego algorytmu poszukiwania kandydatów na prototyp oraz zoptymalizowanego (o mniejszej złożoności obliczeniowej) algorytmu poszukiwania kandydatów na prototyp.

Istotny element składowy rozprawy stanowi implementacja w systemie komputerowym zaproponowanej strategii uczenia i opracowanych modeli szczegółowych, a następnie przeprowadzenie szeregu symulacji z użyciem powyższej implementacji oraz na bazie przykładowych zbiorów danych dla uczenia maszynowego (repozytorium *Machine Learning Repository*, *University of California, Irvine*).

W podsumowaniu rozprawy zaproponowano przyszłe kierunki badań i rozwoju modelu systemu agentowego w obszarach: modelu wiedzy agenta (obejmującego sposób reprezentacji kategorii z prototypem); strategii uczenia kategorii z prototypem; modelu semantyki kognitywnej; oraz wydajnościowej optymalizacji modeli implementacyjnych.

Abstract

In the dissertation, a model of an agent system was developed and verified that processes the knowledge about the membership of objects to the category with the prototype and communicates this knowledge using modal epistemic statements (in the first person). In particular, a class of epistemic statements built of modal operators of knowledge (*I know that ...*; *I'm sure that ...*), strong beliefs (*I believe that ...*) and weak beliefs (*I find it possible ...*) is considered. Statements of this type should constitute an important group of language messages processed by interactive agent systems.

The main research goal accomplished in the dissertation was to prove the thesis that it is possible to design and effectively implement an agent system capable of processing the considered group of language messages in accordance with common-sense postulates reflecting the rules of processing this class of statements in natural language.

The original and extending the current state of knowledge achievement presented in the thesis is the development and verification of a model for processing epistemic modal statements about membership of objects to a category for the case of a category with a prototype. The implementation of the task required reference to the theory of categorization formulated in the field of cognitive linguistics, in particular to the so-called standard version of prototype semantics.

The demonstration of the main thesis of the dissertation required the implementation of three detailed research objectives: the development and verification of the agent's knowledge model, the development and verification of the model of the so-called cognitive semantics of modal epistemic statements and the development and verification of exemplary implementation models.

The development of the agent's knowledge model included: defining the internal representation of world objects and the original way of processing them in the mental space of the agent system; defining a category model with a prototype (including the core, boundary, and outer region of the category); and developing an overall strategy for training a category model with a prototype. The verification of the properties of the knowledge model and the process of learning the category with the prototype consisted in proving that the category model obtained as a result of applying the developed strategy maintains the correct relationship between the core, boundary, and outer region of the learned category, and in analytical determination of the computational complexity of the developed learning strategy.

The development of a model of cognitive semantics of modal epistemic statements included: defining the syntax of a modal categorization language that expresses the knowledge of the agent system about the membership to a category using modal operators of knowledge, strong beliefs, and weak beliefs; defining the concept of epistemic neighborhood; adaptation, for the purposes of cognitive semantics defined in the dissertation, of the concept of relative grounding strength of modal statements introduced in the the-

ory of modal grounding of the language of communication for the case of categories without a prototype; a detailed discussion of the postulated method of applying the cognitive semantics of modal statements about the membership of an object to a category with a prototype; and the development of a set of the epistemic satisfaction relations of modal formulas representing individual cases of cognitive semantics of modal epistemic statements about the object's membership to a category with a prototype in versions for objects with a complete and incomplete representation of their state. The verification of the originally proposed model of cognitive semantics consisted in presenting proofs of twelve theorems about the compliance of the developed cognitive semantics with the pragmatics of using modal statements in natural language about the object's membership to the category with a prototype, and discussing the relationships of the proposed cognitive semantics with the theory of grounding the modal language of communication in agent systems, formulated for the case of the category without a prototype.

Three implementation models were proposed and verified in the dissertation: a model based on an unweighted Hamming distance, a model based on a weighted Hamming distance, and a model based on an asymmetric weighted Hamming distance. For each of the above models, two versions of the search for a set of candidates for a prototype – centroids or medoids, respectively – were developed. The verification of the implementation models for each case included the proof of the theorem on the computational complexity of the naive prototype candidate search algorithm and the optimized (of lower computational complexity) prototype candidate search algorithm.

An important component of the dissertation is the implementation of the proposed learning strategy and developed detailed models in a computer system, and then conducting a number of simulations using the above implementation on the basis of sample data sets for machine learning (*Machine Learning Repository*, *University of California, Irvine*).

In the summary of the dissertation, future directions of research and development of the agent system model were proposed in the following areas: agent knowledge model (including the method of representing categories with a prototype); a category learning strategy with a prototype; cognitive semantics model; and optimization of performance of implementation model.

Contents

Symbols

$\models_G$	symbol of epistemic satisfaction relation
λ_c	relative grounding strength
λ_{minBel}	threshold indicating minimal relative grounding strength needed to support Bel operator
A	set of attributes
AWH	Asymmetric Weighted Hamming distance
BGM	Basic Grounding Model
$Bel(\varphi)$	modal operator with assigned meaning "I believe that φ"
$cReps$	function returning a set of objects with complete information for object with incomplete information, i.e. $ExtO \longrightarrow \Pi(O)$
$E_c^+(Exp_c)$	set of all mental representation of objects at least once confirmed as c in the learning experience Exp_c
$E_c^-(Exp_c)$	set of all mental representation of objects at least once rejected as c in the learning experience Exp_c
$\hat{E}_c^+(Exp_c)$	multiset constituting a quantitative reflection of a cumulative experience gathered in Exp_c and related to objects being confirmed as c
$Entropy(S)$	entropy of collection S
$Episode(t)$	formal representation of episode in time point t
$ER(\tau_c^-, \tau_c^+)$	function calculating epistemic radius based on radii τ_c^- and τ_c^+
$EN_c(o, \varepsilon)$	epistemic neighborhood of object o with radius ε in model m_c
$ExtEpisode(t)$	formal representation of extended episode in time point t
$ExtO$	universe of mental representations of distinguishable objects with incomplete information

$extPercept$	function representing observed values of attributes of objects, $X_t \times A \longrightarrow V$, such that $percept(x, a) \in V_a \cup \{\epsilon\}$ for all $x \in X_t$ and $a \in A$
$extractCandidates$	function $\hat{\Pi}(O) \rightarrow \Pi(O)$ representing method of determining a set of candidates for a prototype
f_{AWH}	Asymmetric Weighted Hamming distance function
f_H	Hamming distance function
f_{WH}	Weighted Hamming distance function
$Gain(S, a)$	information gain of an attribute a relative to collection of objects S
Exp_c	multiset quantitatively aggregating an overall learning experience related to a concept c
is-c	positive label indicating category c
$Know(\varphi)$	modal operator with assigned meaning "I know that φ"
$label$	function representing observed label to object assignments, $X_t \longrightarrow \Pi(L)$
L_t	finite set of labels in $Episode(t)$
m_c	cognitive model of category c
macrostructure	function $f : O \times O \rightarrow \mathbb{R}^+ \cup \{0\}$
M	set of models of categories
not-c	negative label indicating category c
$(o, +)^c$	mental representation of a piece of a positive learning experience
$(o, -)^c$	mental representation of a piece of a negative learning experience
$P(c \mid a = v_a)$	probability that object belongs to category c under the condition that value of attribute a is equal to v_a
$o_c^\star$	prototype of category c
O	universe of mental representations of distinguishable objects
$percept$	function representing observed values of attributes of objects, $X_t \times A \longrightarrow V$, such that $percept(x, a) \in V_a$ for all $x \in X_t$ and $a \in A$
$Pos(\varphi)$	modal operator with assigned meaning "It is possible that φ"
$PS(t)$	t-related state of cognition of agent $PS(t) = \{M, Episode(t)\}$

$PSE(t)$	extended t-related state of cognition of agent $PSE(t) = \{M, ExtEpisode(t)\}$
$\Pi(X)$	power set of X
$\hat{\Pi}(X)$	set of all multisets defined over the set X
S_t	set of statements generated by agent in time point t
t_i	point in time
T	set of points in time
τ_c^-	radius of the boundary of category c
τ_c^+	radius of the core of category c
V_a	set of values for attribute a (domain of a)
WH	weighted Hamming distance
X_t	finite set of objects in $Episode(t)$
$\hat{X}$	multiset X

Chapter 1

Introduction

1.1 Motivation

It is natural that human beings create tools to help them with everyday tasks. However, as the tools get more complex, a special training might become necessary to apply the tools and attain desired results. Current artificial systems are extremely complex and knowledge about specifics and technical intricacies of their insides goes beyond understanding capabilities of a typical user of such systems.

A multidisciplinary field of Human-System Interaction (or Human-Computer Interaction) aims to cross the chasm between a user and a complex technical system by creating means of communication between these two otherwise incompatible beings. As a result of years of research in the field[12, 29, 67], multiple abstraction layers and intuitive means of communication have been introduced. They allow users to perform complex technical tasks without fully understanding the detailed technical behavior of the machine.

Due to the human nature, natural-language-based interfaces are one of the most natural directions of the undertaken research. It might look like (and it is, in all honesty) a huge overhead to add natural language capabilities to a machine that could be otherwise controlled with just a few buttons. Still it seems to be a generally accepted desire to be able to verbally ask a machine to "*make you a large latte*" rather than being forced to memorize a particular button combination that leads to the very same outcome. The key justification point (and, at the same time, the key design goal) for such an approach is that users are allowed to communicate freely in unconstrained manner and that natural language interfaces require little-to-none additional training in order to be used.

This convenience of using ordinary speech is why the field of artificial intelligence so often considers applications of the natural language in artificial systems. However, the issue is extremely complex due to multiple reasons. Natural languages – at least those known to the author – are characterized by a wealth of means of expression. Moreover, humans using natural language refer to their contextual and subjective knowledge, i.e., to a way they perceive, process, categorize, understand, and describe the environment ("*Robo, quickly bring that twirly toy to the crying child.*") and to their internal states by forming modal epistemic statements ("*I believe this approach might be unsuitable in our situation.*").

The complexity of natural language directly translates into high levels of complexity of models describing generation of even the simplest statements of natural language. The natural language processing systems (NLP) developed in recent years[9, 14] need hundreds of billions of parameters to work, the values of which are obtained in computational processes using thousands of petaflop/s-days[1]. They have been significantly successful, but they also fail in some areas[6], and their main disadvantage is the inability to predict and explain wrong or undesirable results of their own processing. Hence, e.g. the authors of the GPT-3 system allow a wider audience to use the API for the above system only under their supervision[8].

The above approaches – usually based on artificial neural networks – are often said to behave like black boxes[27, 97]. This requires pointing out a critical distinction. While, as mentioned before, the user of the system should not be obliged to understand technical intricacies of said system, it is a desired property of the system that its behavior is transparent to its designer and maintainer. Rudin even insists[78, 79] that only interpretable models should be used for high-stake decisions. In a nearby research area of fuzzy-systems, Babuska[3] presents a contrast between approaches based on black-box models and models that are "*transparent to interpretation and analysis*", and justifies an importance of said transparency in modern systems.

Another, nevertheless important, advantage of transparent systems is that they can be analyzed. A natural strive to at least partially understand the way in which humans gather, organize, process, store and convey information[20, 61] translates into the requirement for formal and technical models trying to mimic particular processes and behaviors within humans. Multiple existing approaches to a formulation of cognitive architectures[40, 71, 81] aim to help with questions like: can a particular internal organization of an artificial cognitive system effect in chosen elements of human-like behavior?

One of the approaches that might be considered as an answer to the problem of transparency of natural language models are models based on the cognitive semantics. In cognitive semantics, models aim at describing in detail the relationship between the spoken element of language (word or sentence) and the mental material represented by this element, gathered in the mind of the subject using the language. It is possible to

[1]A petaflop/s-day (pfs-day) consists of performing 10^{15} neural net operations per second for one day, or a total of about 10^{20} operations[2].

give precise technical definition and conditions for the above relationship, which in turn allows to achieve transparent behavior of the system. Nevertheless, cognitive semantics models must take into account various aspects of the mental material, as well as the very manner of relating to this material, which represents the actual object accessible by the agent's perception. Constructing a model of the meaning of even the simplest sentence of a natural language from a theoretical and technical point of view, usually turns out to be very complex due to the multidimensional reference of even simple sentences to e.g. empirical material accumulated over the years. Cognitive linguistics[44] claims that there are no distinct boundaries between language and other psychological competences such as perception, memory, and categorization. Thus, various mental processes that achieve the above abilities must closely cooperate with each other. The realization of the above assumptions is described later in this work.

This dissertation presents the design and analysis of an artificial system capable of generating a limited class of statements in a formal language inspired by a natural language. The above system belongs to the class of cognitive agents[11], that operate (physically or virtually) in a specific environment and interact with humans or other artificial systems. A good example of the above agent can be a program that searches the Internet for interesting data or a robot that moves in an environment that is inaccessible or hostile to humans, e.g. on Mars or in the depths of the oceans. The basic tasks of the agent are as follows:

- obtaining data from the environment,
- obtaining data from other participants of the interaction,
- processing of the obtained data,
- generating statements about the current state of the environment.

The particular methods for carrying out these tasks and the assumptions adopted in this work are described below.

1.1.1 The tasks of the agent

Obtaining data from the environment. It is assumed that an external world of the agent is an environment composed of atomic entities possessing certain features. The entities are dynamic, i.e. their features may change over time. Time is represented in a discrete form, i.e. divided into episodes (snapshots), in which the observations of the entities are available to the agent's perception. The duration of the episode may vary – the most basic assumption is that a single episode corresponds to a period of time during which the values of the features of the observed entities remain unchanged. The above values of features are also discrete. The agent's perception is limited, i.e.

the entities may only be accessible to the agent's perception in certain episodes, as can their features, the values of which may not always be available to the agent.

Obtaining data from other participants of the interaction. Each entity in a given episode can have a label informing the agent to which category the entity belongs or does not belong. We assume the existence of an effective mechanism linking the label with a specific entity. The source of labels is a teacher, so they represent the system of concepts that the teacher uses. This system is related to the language used by the teacher for communication. It has a social character, its form must be agreed on by all participants of the interaction. Therefore, the agent's learning is a supervised learning, preferably with elements of student-teacher interaction, which means that the teacher provides the agent with information that aims to influence and improve the agent's erroneous statements. This is in line with the evolutionary approach used by some language researchers, which assumes that language has evolved by natural selection to reduce mistakes in communication[58]. In this work it is assumed, that the teacher does not have to be one being, but it can be a whole group of subjects using the same language of communication.

Processing of the obtained data. The agent must undergo a learning process, as a result of which it will create internal cognitive structures based on the data received from the environment and from the teacher. These structures allow for the effective use of the externally imposed system of concepts (categories). Difficulties in this regard arise from the following facts:

- the agent did not participate directly in the process of creating the concept system used by the teacher, so e.g. it does not have access to the episodes or learning datasets on the basis of which the teacher created their own concept system; it only has generalized (indirect) information in the form of labels,
- the agent has no access to the teacher's internal cognitive structures,
- the teacher's perception of the world may differ from the agent's perception, e.g. the teacher may have a color camera, while the agent only has a monochrome one.

The cognitive models corresponding to the learned concepts (categories) constitute an ontology built into the agent's cognitive structures. In this work, the cognitive models of categories are considered to be independent from each other. In further research it is planned to expand them with relationships between the models, thus creating a semantic network.

Generating statements about the current state of the environment. The agent should be able to present the results of exploration of the environment in a form that is understandable to other participants of the interaction. In this work, the above results are communicated by the agent as textual (possibly vocalized) statements. The messages generated by the agent are based on its internal structures reflecting the system of concepts used by other participants of the conversation. The agent is also able to determine to some extent the correspondence of its internal structures and the external system of concepts. The above assumptions should have a positive effect on the reliability and usability of the information provided by the agent.

The process of generating statements in relation to the agent's internal cognitive structures is called semantic language grounding[2][31, 84]. The language used by the agent allows to express statements regarding the membership of an object to a category and at the same time determine the degree of certainty of the information provided. In linguistics, the sender's attitude towards the communicated phenomenon, expressed in the structure of the statement, is called the modality[22]. For this reason, the statements generated by the agent are called modal atomic statements. The modal atomic statements that the agent in this model can utter and interpret, are as follows:

- "I know that object x (belongs/does not belong) to category c" – the agent is sure that object x belongs (or does not belong) to category c,
- "I believe that object x (belongs/does not belong) to category c" – the agent has high confidence that object x belongs (or does not belong) to category c,
- "I find it possible that object x (belongs/does not belong) to category c" – the agent considers that object x belongs (or does not belong) to category c, but it is not firm confidence.

The above statements are spoken in the first person, which emphasizes the fact that they express the beliefs of the agent and not the objective state of the outside world. This type of modality is called epistemic modality[45].

The grounded statements must be rational from the point of view of the living (human) participants in the conversation. For example, it is unacceptable for the agent to simultaneously generate statements representing different states of knowledge about the same object and concept, such as "I know that object x belongs to category *bird*" and "I believe that object x belongs to category *bird*".

1.1.2 Inspirations

As already mentioned, the formal language used by the agent is inspired by natural language. It expresses various degrees of uncertainty of the subject generating a state-

[2]Also: symbol grounding.

ment, regarding membership to the category of objects available to the perception of the subject. Since we want these statements to be understandable by the human participants of the conversation, it is worth referring to the suggestions provided by the results of research on human categorization mechanisms and their effects in language. This dissertation refers to the so-called standard version of the prototype semantics developed by Eleanor Rosch[72]. The experiments she carried out showed that in human categorization process, objects belonging to a category are graded, i.e. they may be more or less representative for a given category. The best representative is called the prototype. Its distinction influences the category structure and produces different kinds of prototype effects, which will be described in chapter 3.2.

Hence, the adopted syntax and semantics of the statements reflect the degree of uncertainty related to the representativeness of the object for a given category. In order to correctly implement the generation of statements in accordance with the above pragmatic interpretation, it is necessary to design and implement the appropriate cognitive structures containing an embodied system of conceptual categories with prototypes, or at least allowing for the achievement of prototype effects. The above structures must contain at least a summary of the agent's empirical experience, and possibly also the experiences themselves related to the relevant conceptual categories.

The previous studies[32, 92] have already considered generating statements in formal languages that express uncertainty, including the modal atomic statements. In the mentioned previous stages of the research, the generated statements focused mostly on values of particular physical features of the observed objects, rather than on such abstract concepts as an inclusion of an object to a particular category. A novelty in this work also lies in the reference of statements to the agent's internal structures based on the concept of a prototype. The above structures allow the agent to determine the degree of ambiguity of categorization and to analyze the degree of convergence of the cognitive structure with the model used by the community. Moreover, the agent presented in this work is designed in a way not only allowing it to learn categories (and to categorize further objects into the already-learned categories), but to express the results of the performed categorization while further enriching it with an expression of agent's subjective certainty about the assignment of a particular object to the category. The above abilities translate into greater credibility of the statements generated by the agent.

1.2 Dissertation thesis and research tasks

Thesis *It is possible to design and effectively implement an agent system capable of processing the considered group of language messages in accordance with common-sense* *postulates reflecting the rules of processing this class of statements in natural language.*

Following the assumptions stated above, a set of specific **research tasks** has been defined and tackled within this dissertation:

- development of postulates for the formal representation of cognitive models and for providing their technical definition,
- development of a method for generating the embodied ontology in the scope covering a built-in representation of a system of conceptual categories with a prototype,
- development and codification – in the form of a set of postulates – of the conditions of permissible and unacceptable use of sets of modal statements about the membership of an object to a category,
- definition of the original cognitive semantics of the language of modal statements about the membership of an object to a category for a class of agent systems with an ontology learning module,
- proposal and analysis of specific numerical models that perform the above tasks,
- simulations based on the selected methodology of implementation of the agent and the implementation of cognitive semantics.

The original element of the work is to conduct a study, mainly based on literature studies, leading to the development of a list of obligatory structural and procedural components of the management module for this part of the agent's embodied ontology, which is responsible for representing the agent's knowledge about the scope and structure of conceptual categories with the prototype. Setting such a goal as one of the main theoretical problems considered in the dissertation results from the utilitarian assumption, according to which the developed mechanisms of grounding of atomic statements are to refer to a wide spectrum of real artificial agent systems, i.e. they must take into account the fact that they are created using a variety of methodologies and implementation tools, processes and knowledge representations. The developed general architecture is to constitute a metastructure that can be mapped (projected) onto systems with different knowledge representation mechanisms. In other words, as part of the dissertation, a certain system of structures is also developed, the implementation of which in various contexts is to lead to the creation of an artificial agent with the competence to express in a semi-natural language the belonging of the observed object to one or more conceptual categories with the prototype.

In this sense, the proposed considerations are of a research and development nature.

1.3 Components of dissertation

The text of the monograph is divided into eight parts: an above introduction, incoming motivation, research background, four chapters dedicated to particular theoretical and design issues, and a summary. For clarity, the text is supplemented with lists of tables and figures, and a list of the cited literature.

The second chapter contains one of the possible scenarios that motivate the research described in this work. It presents a practical situation in which an autonomous agent operates in an environment that is difficult for humans to access and communicates the results of exploration of the environment in a way that is understandable to human recipients.

The third chapter presents the areas of cognitive science covered by this work and the results of research to date in these areas. In particular, there are discussed the theories of human-conducted categorization with a prototype, and the theory of grounding along with the cognitive semantics of modal atomic statements in artificial systems on the subject of exhibiting or not exhibiting the given property by some object – so-called *Basic Grounding Model*.

Chapter four contains a general model of a category learning strategy with a prototype. It presents the model of the external world, the agent's internal cognitive structures and the way of constructing these structures based on the experience gathered by the agent. The cognitive model representing the category contains a prototype and is divided into three areas representing a different degree of certainty regarding the belonging of the object to the category. The chapter presents an algorithm that allows the selection of the prototype and the division of mental space into the above areas, but without the method of determining candidates for a prototype, which was specified in chapter six. Chapter four also contains a computational example and scheme for evaluation of computational complexity of strategy.

Chapter five defines the cognitive semantics of modal statements about the belonging of an object to a category, referring to the cognitive models presented in the previous chapters. First, the semantics for complete representation of an object is presented, and then its extended version for incomplete representation of object, that is, for situations where not all values of the object's features are known. The formal language of modal atomic formulas and the conditions for generating individual formulas were defined. Moreover, some desirable properties of semantics regarding the simultaneous generation of modal atomic statements have been proved. Finally, the relationship between the model described in this dissertation and the cognitive semantics of modal atomic statements on exhibiting or not exhibiting the given property by some object is presented.

The sixth chapter contains three proposals of implementation models that constitute a specificity of the ideas presented in chapter four, and thus define the mechanism of selecting candidates for a prototype and the so-called macrostructure. For each of the models, an analysis of computational complexity and an optimized version of the candidate selection algorithm were presented.

The seventh chapter presents the simulation results for the data downloaded from the Machine Learning Repository, University of California, Irvine. The above simulations prove the technical feasibility of the model.

The eighth chapter summarizes the doctoral dissertation. It provides a list of theoretical and practical problems discussed in previous chapters and formulates directions for further research and implementation work.

Chapter 2

Motivation scenario

We consider a situation that is frequent in practice, when the system is to operate autonomously, without constant human supervision. In this scenario, it is the agent managing a group of robots working on Mars. Their task is to explore the area, but due to the communication delay, they cannot be monitored by humans in real time. That is why they are monitored by a designated artificial agent, whose task is to warn when the robot is in a dangerous situation. In classic automation systems, operating e.g. in factories, the danger is usually determined by a set of strictly defined alarm signals related e.g. to a specific position of a limit switch or exceeding a certain level by an analog indicator such as pressure or temperature[88]. Such a system is able to detect only situations which occurrence has been anticipated and implemented, and in fact it happens that this approach fails, resulting in the occurrence of life and property threatening failures[57]. Knowing this, human factory operators attach great importance to direct supervision of computer systems, also by means of physical inspection to obtain current environment's state[56]. In a situation of exploring a distant planet, it is even more difficult to predict all possible configurations of variables, and the very concept of a "dangerous situation" is inherently fuzzy and difficult to define. Therefore, in this case, it seems to be a better solution to teach the agent the sense of the above concept using examples. After the training process is completed, this will allow the agent to generate messages regarding the different level of threat to individual robots.

In such applications, we must determine what cognitive potential of the agent is necessary to achieve the assumed goals. In this case, the agent is embedded in a specific reality in which it has the ability to recognize physical entities. Establishing the existence of an entity is based on its location in time and space. In addition, the agent must be able to assign a specific set of features to a separate entity and, possibly, associate a given entity with the label given to it by the teacher, which is a person or a group of people overseeing the mission from Earth.

Suppose the agent is able to determine three attributes (features) for each robot:

- *Battery* – its value is *low* if robot's battery is close to exhausted, *ok* otherwise,
- *Sunny* – its value is *yes* if the robot is in a sunny place which, among other things, allows its batteries to be charged by photovoltaic panels; otherwise its value is *no*,
- *Terrain* – the type of terrain on which the robot moves, it takes the following values: *sand*, *rock* and *gravel*.

The exemplary case is shown in the figure 2.1. Within the agent's perceptual range, there are four robots that move on different types of terrain. The darker (upper right) part of the figure represents the shaded area where the robots cannot charge their solar cells. The agent is in the base where robots can go to recharge their batteries or repair. To simplify the example, we will mainly consider the danger that the robot will get stuck outside the base with a discharged battery, which will prevent it from further operation. Of course, in a real scenario, we expect the agent to be able to detect more types of danger, based on more attributes.

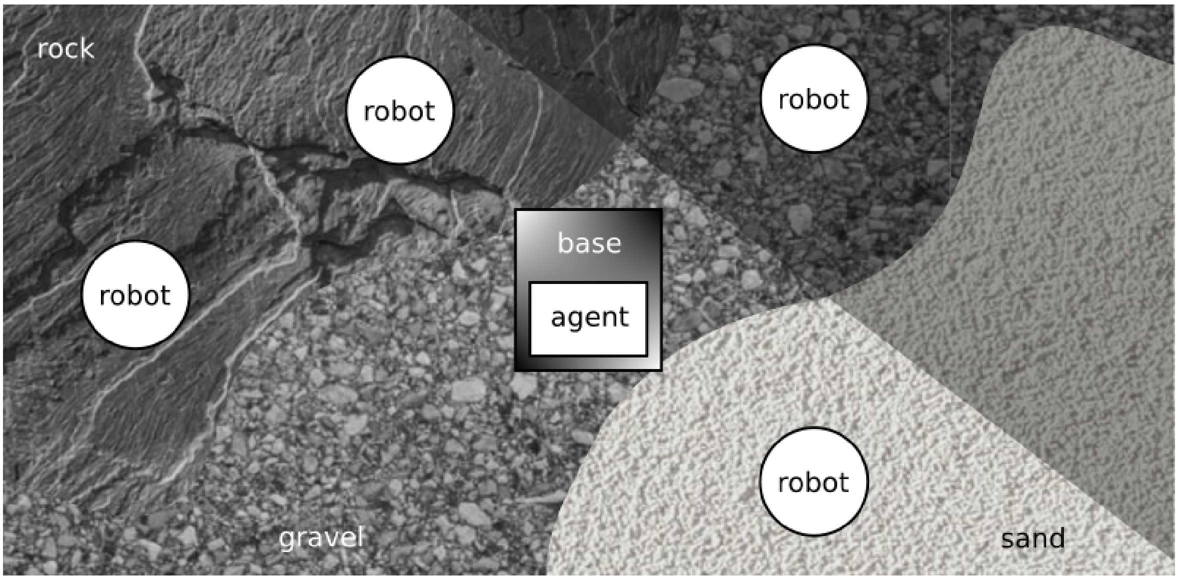


Figure 2.1: Robots in agent's perception perimeter.

First, the agent must separate the four entities e_1, ..., e_4 from the surrounding world, corresponding to the four robots. The result of this process is the construction of four objects $x_{1,1}$, ... $x_{1,4}$ in the agent's working memory. Since the object model, in addition to the fact of separation in the surrounding world, also specifies the recognized states of attributes, the agent determines the values of attributes for each of the entities. The figure 2.2 shows the mental state of the agent in such a situation[1]. The figure shows the state of four robots in episode no. 1:

[1]The figure uses abbreviated names of attributes and their values: B=*Battery*, l=*low*, o=*ok*, S=*Sunny*, y=*yes*, n=*no*, T=*Terrain*, s=*sand*, r=*rock*, g=*gravel*.

- the robot represented in the agent's working memory by the object $x_{1,1}$ has a charged battery ($Battery = ok$), is located in a sunny place ($Sunny = yes$) and moves on rock ($Terrain = rock$),
- the robot represented in the agent's working memory by the object $x_{1,2}$ has a charged battery ($Battery = ok$), is located in a sunny place ($Sunny = yes$) and moves on rock ($Terrain = rock$),
- the robot represented in the agent's working memory by the object $x_{1,3}$ has a battery that is almost drained ($Battery = low$), is located in a sunny place ($Sunny = yes$) and moves on sand ($Terrain = sand$),
- the robot represented in the agent's working memory by the object $x_{1,4}$ has a battery that is almost drained ($Battery = low$), is located in a shadow ($Sunny = no$) and moves on gravel ($Terrain = gravel$).

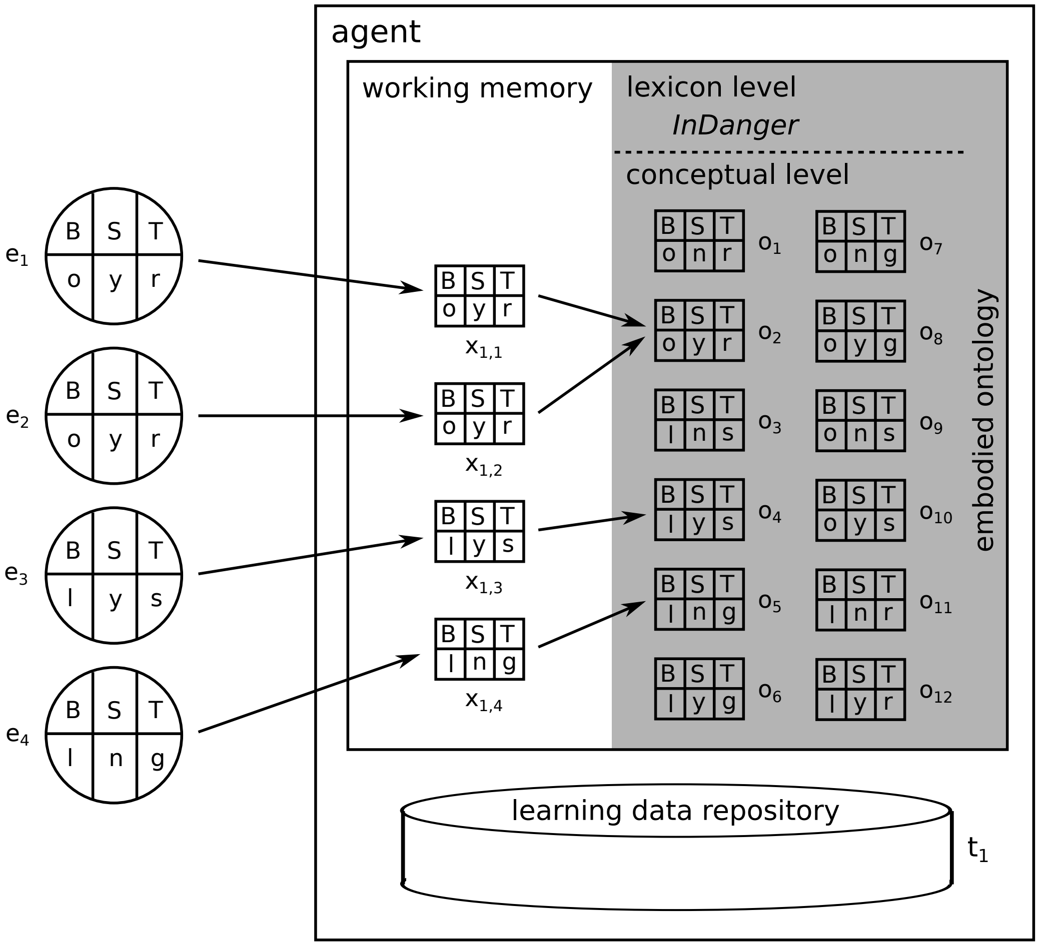


Figure 2.2: Mental representation of episode no. 1.

Objects x_{ij} from working memory are projected to o_k objects in the agent's embodied ontology. It may happen that two x_{ij} objects will have the same set of features, except for their location in time and space, of course. This situation means that the recognized x_{ij} objects are cases (realizations) of the concept of an individual object o_k. For example, in the figure 2.2 the objects $x_{1,1}$ i $x_{1,2}$ are realizations of the individual object o_2.

Reference to the ontology allows to ground potential messages generated by the agent. Messages that meet the conditions of proper grounding are passed by the agent to the recipients, in this case the people supervising the entire mission from Earth. Suppose that in episode no. 1 the agent generated the following statements:

- "I know that object $x_{1,1}$ does not belong to category $InDanger$",
- "I know that object $x_{1,2}$ does not belong to category $InDanger$",
- "I believe that object $x_{1,3}$ belongs to category $InDanger$",
- "I find it possible that object $x_{1,4}$ belongs to category $InDanger$".

We consider a situation where the agent is still in the learning phase, i.e. the teacher supervising it believes that the agent has not yet fully learned the concept of the category $InDanger$. The teacher, analyzing the agent's statements and the features of the robots they refer to, states that the statements about $x_{1,3}$ and $x_{1,4}$ objects do not meet the teacher's expectations, so it is worth giving labels to robots that will help in better comprehension of the meaning of the concept by the agent. In episode no. 2 – shown in the figure 2.3 – the relevant entities are thus labeled by the teacher:

- "*is-InDanger*" indicating that entity e_3 belongs to the category $InDanger$,
- "*not-InDanger*" indicating that entity e_4 does not belong to the category $InDanger$.

Associating an entity with a label updates the agent's cognitive structures. Its knowledge is based on the sequence of the learning situations and is included in the embodied ontology. After updating the cognitive structures, the agent generates messages describing, according to it, the state of the robots in episode no. 2:

- "I know that object $x_{2,1}$ does not belong to category $InDanger$",
- "I know that object $x_{2,2}$ does not belong to category $InDanger$",
- "I know that object $x_{2,3}$ belongs to category $InDanger$",
- "I know that object $x_{2,4}$ does not belong to category $InDanger$".

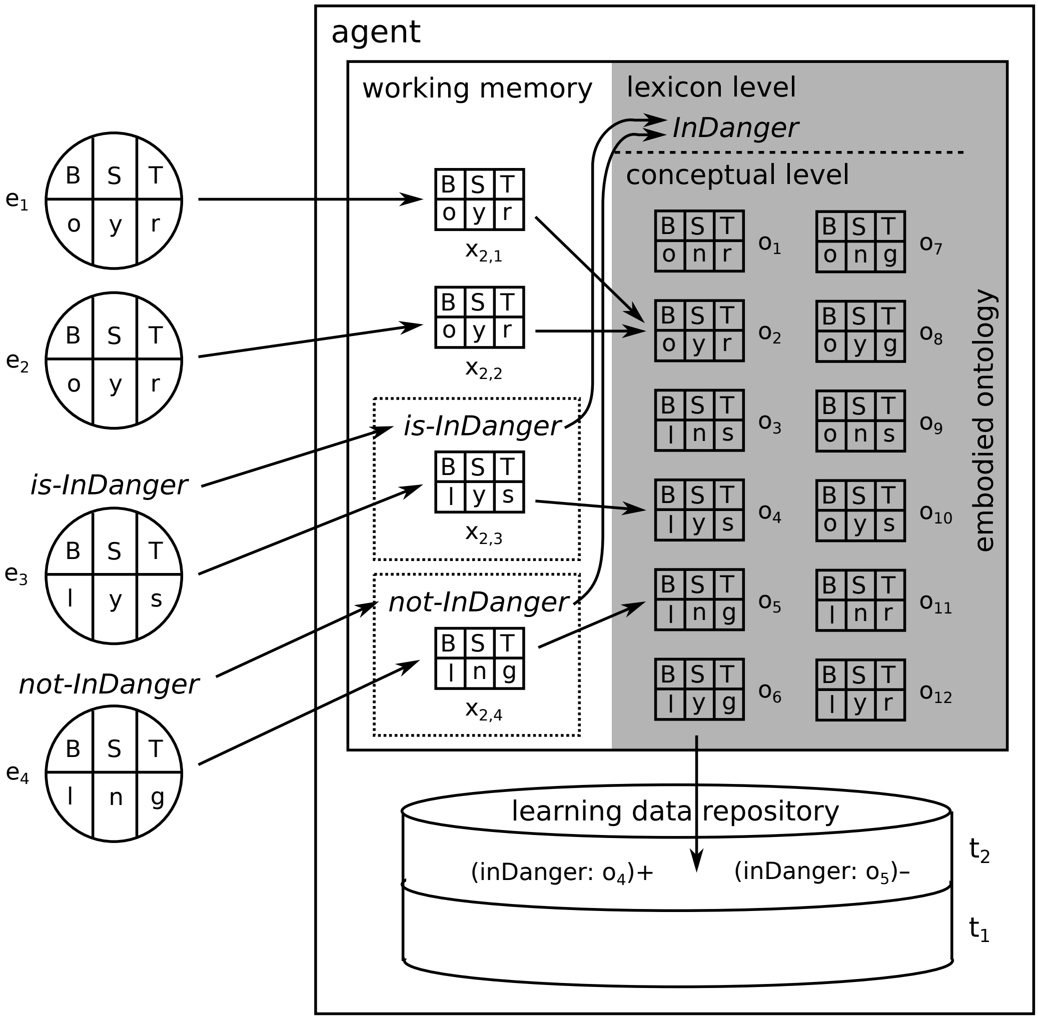


Figure 2.3: Mental representation of episode no. 2.

As one can see, the confidence has increased that the object $x_{2,3}$ representing the entity e_4 belongs to the category $InDanger$. Instead of the modal operator "*I believe*" expressing a strong belief in belonging to a category, the agent used the operator "*I know*" to express complete certainty. In the case of the object $x_{2,4}$ representing the entity e_3, the agent expressed confidence that the object did not belong to the $InDanger$ category, while in the episode no. 1 for the same object it allowed for the possibility that the object belonged to the above category.

The situation corresponding to the episode no. 3 is shown in the figure 2.4. As one can see, the entities e_2 and e_3 have disappeared from the agent's perception field, but the entity (robot) e_5 has appeared in it. In addition, the entity e_1 moved so that it is now in the shaded zone, therefore the attribute value *Sunny* for its corresponding object $x_{3,1}$ has changed to *no*. The transition of the value of the attribute causes the entity e_1 to now correspond to another object in the agent's mental space, namely o_1 instead

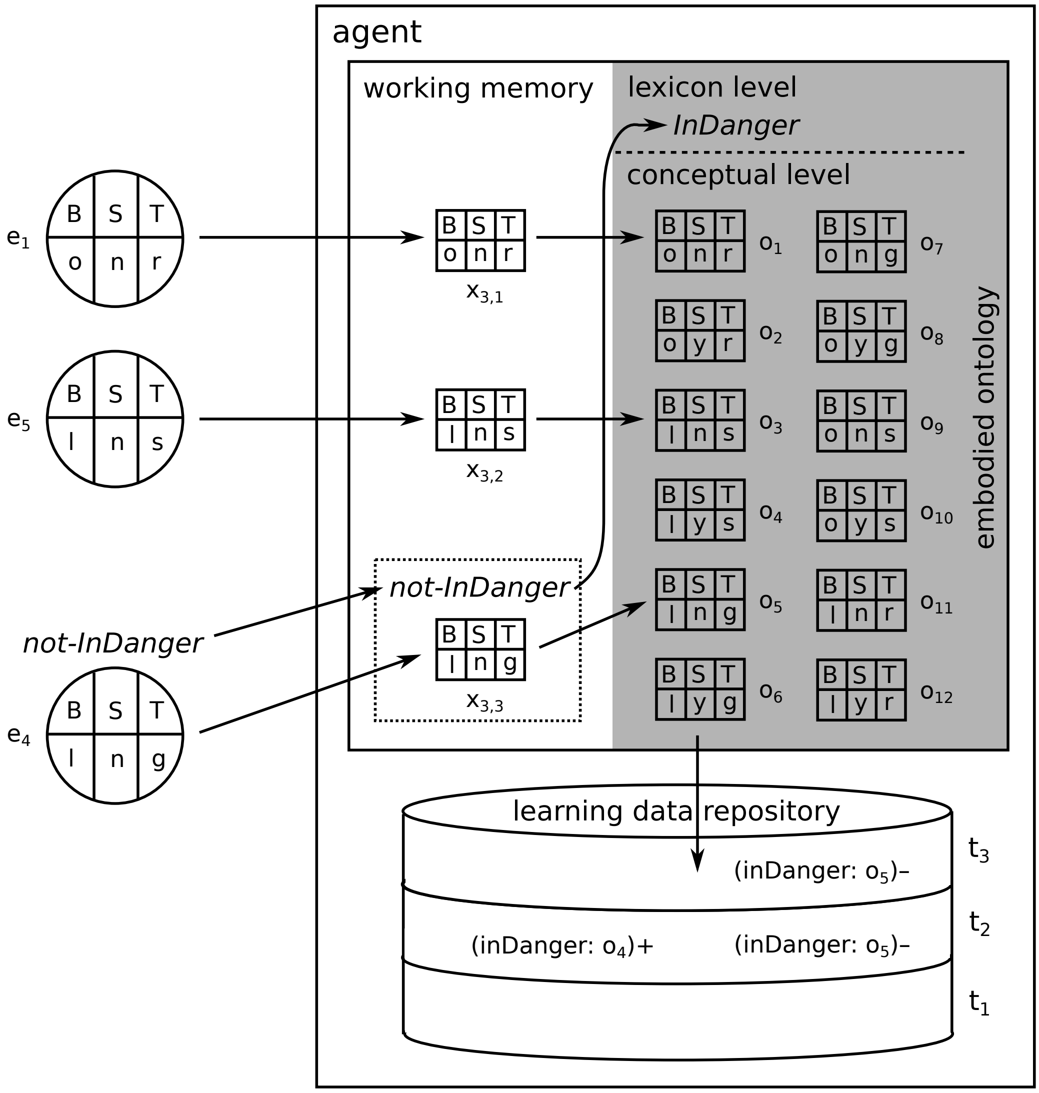


Figure 2.4: Mental representation of episode no. 3.

of o_2 as before. The teacher re-labeled the entity e_4 with the label "*not-InDanger*" indicating that e_4 does not belong to the category $InDanger$.

After the observed objects are related to the cognitive structures, the agent generates the following messages:

- "I believe that object $x_{3,1}$ does not belong to category $InDanger$",
- "I find it possible that object $x_{3,2}$ belongs to category $InDanger$",
- "I know that object $x_{3,3}$ does not belong to category $InDanger$".

The messages for objects $x_{3,1}$ and $x_{3,2}$ do not contain the modal operator "*I know*" as they correspond to combinations of attribute values that are new for the agent. The teacher can re-evaluate their rationality and on this basis try to provide more knowledge about the *InDanger* category by using the appropriate labels assigned to the entity.

In subsequent episodes, the teacher-agent interaction may be similar. When the teacher decides that the agent has learned the *InDanger* category correctly, they may stop verifying the agent's statements and continue with just receiving warnings from the agent. Thanks to this, the system consisting of the agent and the robots observed by it can operate autonomously.

In addition, we assume that while in the learning phase the agent has access to full information on all states of the features of each robot, in the autonomous phase it can generate statements based on incomplete information, i.e. the values of some attributes describing the robot may not be available to the agent's perception. Such a situation is presented in the figure 2.5, where the agent is not able to determine the value of the *Sunny* attribute for the entity e_1 to which the object $x_{n,1}$ in the working memory corresponds. The object $x_{n,1}$ can be the realization of the object o_1 or o_2 from embodied ontology, but the agent is unable to determine exactly which. For this reason, the agent has to consider more options, which can increase uncertainty about the object's belonging to the *InDanger* category. In the given example, the agent signals the above uncertainty by using the belief operator:

- "I believe that object $x_{n,1}$ does not belong to category *InDanger*".

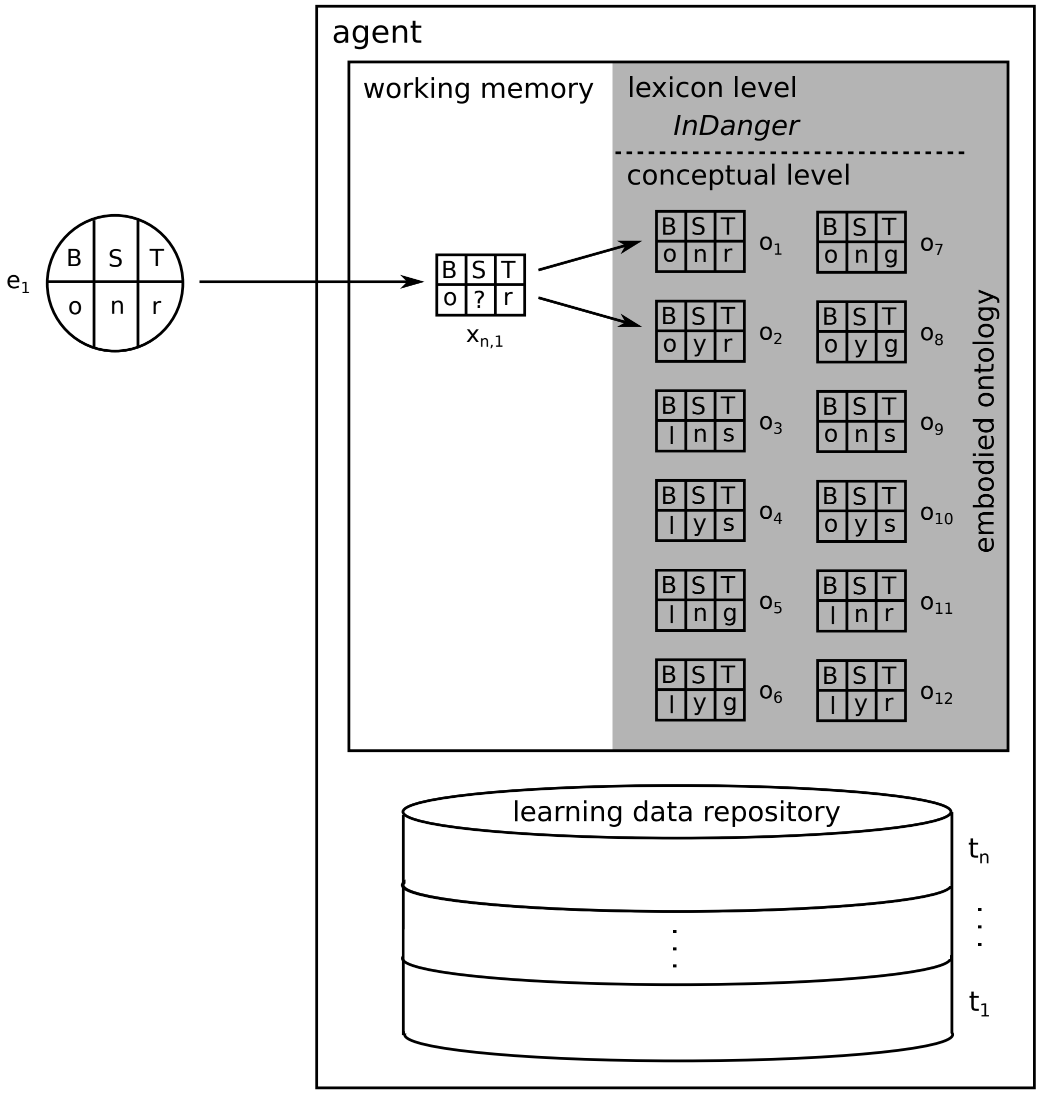


Figure 2.5: Mental representation of episode no. n.

Chapter 3

Models related to the cognitive semantics of atomic statements and the theory of prototypes

As outlined in the previous chapters, this dissertation deals with the problem of development of an artificial agent with embodied cognitive models allowing to generate modal statements about the membership of objects to categories. This chapter discusses the theories and research works that are the basis for the answer to the above research problem.

3.1 Cognitive science

This work can be included in the interdisciplinary field of research on cognition known as cognitive science. Usually it is defined[7, 53, 59] as an attempt to synthesize the problems practiced by six traditional academic disciplines: philosophy, psychology, linguistics, computer science (artificial intelligence in particular), anthropology and neuroscience. The figure 3.1 taken from the so-called Sloan report[83] shows the components of cognitive science as vertices of a hexagon. Similarly, the *MIT Encyclopedia of the Cognitive Sciences*[94] recognizes six fields that contribute to the cognitive sciences:

- philosophy,
- psychology,
- neurosciences,
- computational intelligence,
- linguistics and language,

- culture, cognition, and evolution.

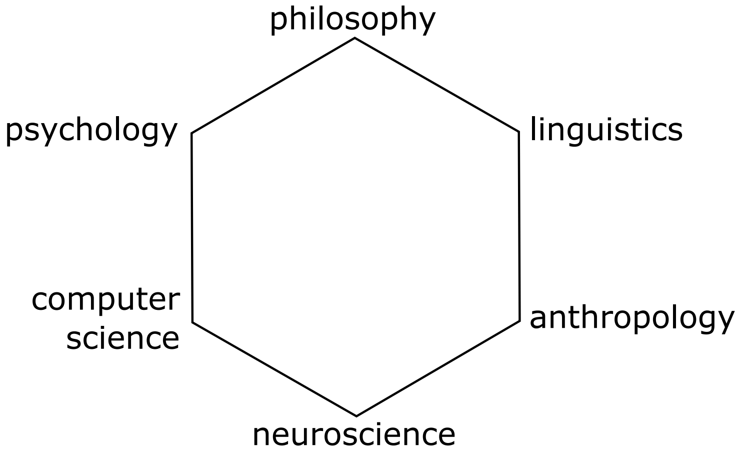


Figure 3.1: Cognitive science.

The research program of cognitive science is not so much about practicing all specific disciplines as about developing or bringing to life those research areas located between the vertices of the hexagon[55]. For example, the line between anthropology and neuroscience represents the research on the evolution of the brain, and by combining neuroscience with linguistics, we get neurolinguistics. The research areas that cover more than two of the six output disciplines are particularly interesting. This work meets this condition because it is an attempt to computationally represent the effects described in the psycholinguistic research. Referring to the figure 3.1 – we operate within the triangle delineated by psychology, linguistics and computer science.

3.2 Theories of conceptual categories with a prototype

Human processing of natural language has long been the subject of research by specialists in philosophy, cognitive psychology and linguistics. It seems beneficial to use their achievements when trying to construct an artificial system that processes natural language or at least adheres to the rules of natural language.

One of the most important aspects of language processing and one of the basic thought processes is the categorization of objects, activities, experiences, etc. Categorization makes it possible to reduce the complexity of the experienced world to a form that can be analyzed by the human mind within a reasonable time. Thanks to this, e.g. when entering a room with a hundred chairs, we do not have to analyze the function of each chair separately. Harnad[24] even states that "cognition is categorization" because people behave in particular ways toward different kinds of things (including abstract concepts such as prime numbers).

The classical theory of categorization was widely recognized for many centuries on the basis of the ideas presented already by Aristotle. It is often referred to as the necessary and sufficient condition model[38] or **criterial-attribute model**[43]. The basic assumptions of the above model are[41]:

- shared properties – there are necessary and sufficient conditions (features) for belonging to a category; each element of a category has all of these characteristics, and there is no element outside the category that has all of these characteristics,
- clear boundaries – it is possible to unequivocally determine whether an item belongs to a category or not; this is in line with the classical set theory,
- uniformity – all elements of the category are equal, there are no more and less important elements; likewise, no distinction is made between the importance of the conditions of belonging to a category,
- inflexibility – the boundaries of the categories do not change.

Research conducted in the second half of the twentieth century in virtually all cognitive science disciplines, showed that in many cases the above assumptions are not met. For example, Geeraerts shows[18], that for the category *bird* it is impossible to find necessary and sufficient conditions for all birds. If we assume that a bird is an animal that is oviparous and has a beak, it turns out that there are species (e.g. a platypus) or even orders of animals (e.g. turtles) that have these features but are not birds. On the other hand, the features that seem distinctive are missing from some elements of the category: ostriches and penguins cannot fly, kiwis do not have wings, and penguins do not have the typical feathers.

Furthermore, the research conducted by Rosch[72] has shown that people do not treat all elements of a category equally. For example, some elements are considered more representative than others. In addition, the experiments showed[74], that for different objects belonging to a category, the categorization time, the time after which the element is categorized in the learning process, and the prototypicality rating obtained from the study participants differ significantly.

Rosch presented a proposal of interpretation of the above effects, called **the standard version of the prototype semantics**. Below, we will discuss its basic theses, listed after [38]:

T1. The category has an internal prototype structure.

T2. The degree of representativeness of a given item corresponds to the degree of its membership to a category.

T3. The elements of a given category do not have properties common to all elements; they are connected by family resemblances.

T4. The boundaries of categories or concepts are fuzzy.

T5. The belonging to a given category is based on the degree of similarity to the prototype.

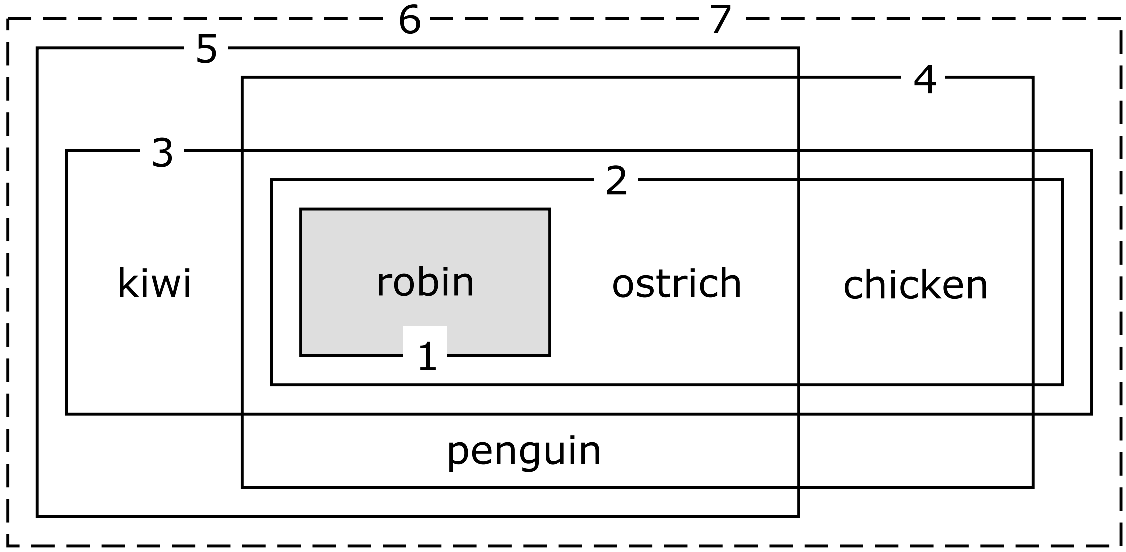
T6. The belonging to a category is not determined in an analytical manner, but in a holistic manner.

Prototype in the above theory (T1 and T2) denotes the most representative instance of a category. As Rosch puts it[73]: "many natural categories are internally structured into a prototype (clearest cases, best examples) of the category with nonprototype members tending towards an order from better to poorer examples". For example, her respondents[72] named apple as the best representative of the *fruit* category. In addition, it was possible to order other items belonging to the *fruit* category on a scale from the most representative to the least representative, they were in turn: plum, pineapple, strawberry, fig and olive. According to the quoted words of Rosch, in the internal structure of the *fruit* category, the apple would be a prototype element, and the remaining fruits would be non-prototypical elements, ordered from the best to the worst examples. Other researchers have also shown that the most representative designates are included in a given category in the shortest time[49, 93] and that new categories are learned faster and with fewer errors when the subjects are shown more typical representatives of the concept[52].

It should be noted that as the research progressed, Rosch herself rejected the thesis that the category must have an internal prototype structure. She recognized the prototype as the result of cognitive processes that can be based on structures built in any way. Thus, various models of these processes can be considered, as long as they demonstrate the prototype effects observed in the research. As Rosch puts it[76]: "prototypes only constrain but do not specify representation and process models". This approach is sometimes called *the extended version of the prototype semantics*[38]. Nevertheless, prototype effects occur during the categorization process by humans and have been confirmed by scientific research.

The question naturally arises as to what determines that an element of a category is considered to be its better or worse representative. Often, the best representative has the most common features with the other items in the category. The exemplary situation is shown in the figure 3.2. It depicts the features of various elements of the category *bird*. The members of the category are described by the names of different species of birds. The drawing was taken from the work of D. Geeraerts[19].

As one can see, the robin has the most common features, and that is why we can



1 - being able to fly 2 - having feathers 3 - being S-shaped
4 - having wings 5 - not domesticated 6 - being born from eggs
7 - having a beak or bill

Figure 3.2: Elements of category *bird*.

consider it as the prototype of the category *bird*. The other elements of the category share different sets of common features. It can be said that the robin functions in the center of the category, and the penguin, for example, in its periphery.

This is an example of a radial structure, not every category has to be compatible with this structure. Wittgenstein[95] noted that there are categories whose elements share common characteristics, but there is no set of attributes that all the elements belonging to the category have. They are connected by the so-called family resemblances (T3), because they resemble each other just like family members who have similar hair, eyes, facial features, etc. As an example of such a category, Wittgenstein gave the term "game". In most games there is an element of competition, but in some (e.g. group dances, role-playing games) there are no winners or losers. The outcome of some games depends entirely on luck (roulette), others partially (poker), some not at all (chess). Some games require physical skills (soccer), some mental skills (chess). It is impossible to provide a set of features that each game has. It is also difficult to give an example of a typical (prototype) game. Moreover, Rosch showed[74], that the more an element is related to other elements of a category by the family resemblance, the faster it is categorized, the faster it is categorized in the learning process and receives a higher prototypicality rating from survey participants.

According to the fourth of the above theses (T4), the boundaries of a category are fuzzy, so it is not possible to clearly say for every element whether it belongs to a category or not. However, for such elements it can be determined to what extent they belong to a given category. This is true for many categories, especially where the features that describe a concept are continuous. A classic example is the analysis of color names by Kay and McDaniel[36] based on the fuzzy set theory[98]. For example, many shades of color can be assigned to the "red color" category, giving each of them a degree of membership from 1 to 0. Let us consider, however, the following sentences, taken from [38], and presenting statements about belonging to the *bird* category with varying degrees of truthfulness:

a) the sparrow is a bird (true),

b) the chicken is a bird (less true than a),

c) the penguin is a bird (less true than b),

d) the bat is a bird (false or very far from the truth),

e) the cow is a bird (completely false).

The first three statements are certainly true, although there is some feeling that some are less true than others. The statement e is definitely false. The statement d confirms the fuzziness of the category boundary, the bat may or may not be included in the *bird* category, depending on the criteria adopted by the subject communicating the sentence or the knowledge of the subject. For example, a person who sees a bat for the first time may classify it as a bird, due to the fact that it has three of the seven features shown in the figure 3.2 (being able to fly, having wings, not domesticated). Only the provision of further information on the method of reproduction and upbringing of offspring by bats, may result in including them in the *mammal* instead of the *bird* category. In conclusion, for the above example **three areas that define the structure of a category** emerge: containing objects that definitely belong to it, containing objects that definitely do not belong to it, and "uncertain" objects that belong to the category boundary. This division will be reflected in the computational structures presented later in the dissertation.

According to the fifth of the above theses (T5), belonging to a given category is based on the degree of similarity to the prototype. Thus "the best examples of a category can serve as reference points in relation to which other category members are judged"[73]. It follows that the categorizer must be competent to compare the elements of the category with the prototype.

The last of the above theses (T6) says that belonging to a category is determined holistically, not analytically. This means that it is not resolved by some specific set of features, as in the case of the necessary and sufficient conditions model, but by the similarity to other elements belonging to the category. Of course in an artificial system, the comparison must, however, be computable.

3.3 Theory of grounding

One of the tasks of the agent in this work is to generate statements about the categorization of an object. An important problem is how in an artificial system the above statements are related to the representations of concepts in the embodied ontology of the agent. This leads directly to a problem of symbol/language grounding[90].

The problem of language grounding belongs to a broad class of basic research problems solved in the field of artificial intelligence and cognitive science. The grounding should be understood as the product of knowledge processing mechanisms, taking place in the agent (subject of knowledge), which are responsible for establishing and maintaining the semantic relationship between the material signs of language and the objects of the external world. The function of grounding models is to explain what form these relationships take for particular agent types and particular communication languages.

Harnad[23] defines the grounding problem by asking questions: "How can the semantic interpretation of a formal symbol system be made intrinsic to the system, rather than just parasitic on the meanings in our heads? How can the meanings of the meaningless symbol tokens, manipulated solely on the basis of their (arbitrary) shapes, be grounded in anything but other meaningless symbols?". Harnad states that cognition cannot be just symbol manipulation. The symbol of a concept (a sign in a given language) must be related to the meaning of the concept, where by meaning it means structures representing the concept in a non-symbolic way. These structures should be based on the agent's perception, i.e. information that is a projection of objects and events through the agent's sensors (senses). Harnad also suggests to start from the simplest representations, and then to build higher-order structures on them, allowing for the categorization of objects. This approach is in line with the one adopted in this dissertation.

The above concepts are often illustrated using the so-called semiotic triangle. It is a concept derived from the field of semiotics[60], but it has also come to be used to describe symbol grounding in artificial systems[77, 89]. The classic semiotic triangle is presented in the figure 3.3.

The referent can be a physical object in the real world, but also an abstract idea or some other sign of language. The thought is also called a reference or an interpretant. So it is interpretation of the referent, a reference that is directed and organized[60]. The symbol is used to communicate according to an arbitrarily adopted convention. The most frequently considered symbols are words in graphic or sound form.

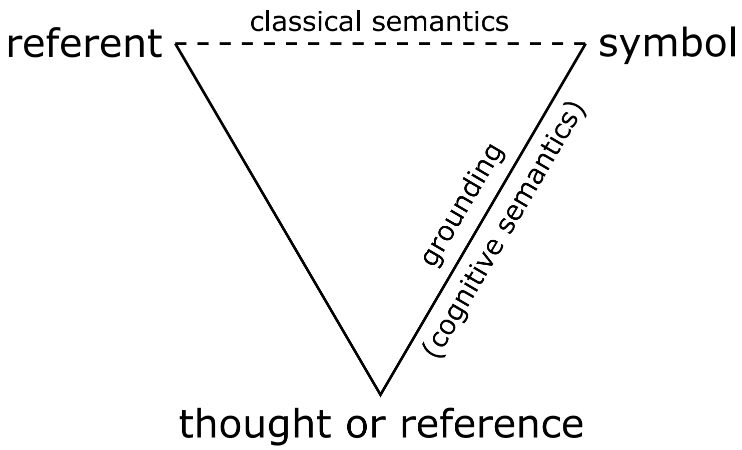


Figure 3.3: Semiotic triangle.

As shown in the figure, the grounding process relates to the relationship between the symbol and the thought. It is defined by **cognitive semantics**, which decides how this process should execute.

Classical semantics refers to the relationship between the symbol and the referent and is marked in the figure by a dotted line. In fact, however, such a relationship does not exist, because for example, there is no direct relationship between the word "dog" and the real objects that we call it. That is why in this dissertation such an approach is considered to be a too far-fetched simplification.

In this work, it is assumed that the thought is placed in the cognitive structures of the artificial agent system. The referent is a physical entity, external to the agent system. The entity's features are available to the agent through appropriate receptors, so the entity is reflected in the agent's working memory as an object which is a combination of the values of the features, obtained using the agent's perception tools. The symbol is a language formula that expresses a statement about the membership of the object to a category (class) of entities. The figure 3.4 presents the above assumptions within a semiotic triangle.

The syntax of the language used by the agent in this model will reflect the relationships outlined above. This means that the agent's statements will not be simply relating to the (allegedly) objective state of affairs. Instead, the formulas generated by the agent will use operators expressing the modality of the statement, i.e. conveying the agent's attitude towards the communicated content. In philosophy and linguistics, many types of modality are considered, and the authors use different divisions of them[22, 45, 66]. For example, in [62] the following main types of modality are distinguished:

- epistemic – where speakers express their judgements about the factual status of the proposition,

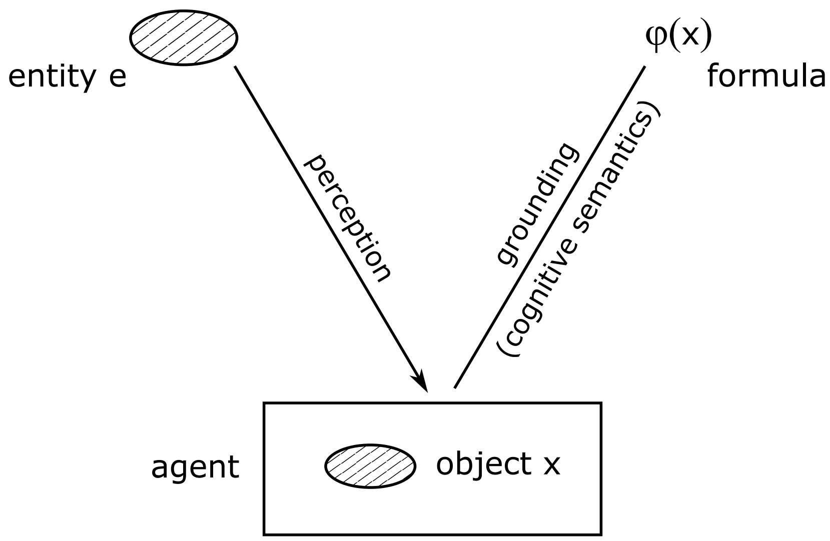


Figure 3.4: Semiotic triangle for agent system.

- evidential – where speakers indicate the evidence they have for factual status of proposition,
- deontic – where conditional factors are external to the relevant individual,
- dynamic – where conditional factors are internal to the relevant individual,
- other – presupposed propositions, negative, interrogative, wishes, fears, etc.

The statements generated by the agent in this model enable the expression of an epistemic modality. The agent uses three different operators, which express different levels of confidence in its judgment. This is in line with the division presented in linguistics[22], where:

- the highest degree of certainty is expressed by declarative statements, also known as assertions; in this model, such a formula is started with the $Know$ operator and its meaning is as follows: "I know that object x belongs to category c" or "I know that object x does not belong to category c",
- a lower degree of certainty is expressed by strong hypothetical statements; in this model, such a formula is started with the Bel operator and its meaning is as follows: "I believe that object x belongs to category c" or "I believe that object x does not belong to category c",
- the lowest degree of certainty is expressed by weak hypothetical statements; in this model, such a formula is started with the Pos operator and its meaning is as follows: "I find it possible that object x belongs to category c" or "I find it possible that object x does not belong to category c".

The above formulas are called the **modal atomic statements**.

3.3.1 Basic Grounding Model

A technically feasible model for grounding the modal communication language in agent systems is shown in [31, 32]. We will refer to this model as **Basic Grounding Model (BGM)**. The above works analyzed the relationships between the internal representations of the agent's individual interactions with the outside world and the mental model generated by the accumulation of these representations, as well as verbally oriented codes of meanings.

In the BGM model, the external world of the agent is a dynamic environment made of atomic objects. The individual states of this world are relativized to points in time. At every point in time t the state of the external world is described with so-called the t-related base profile, that contains:

- the set of all atomic objects located in the external world,
- the set of unique names of properties assigned to objects,
- the information that for a given object at time t a given property has been observed or not; the above information need not to be available for every property of every object (that is, the information may not be complete).

Data for the current base profile are collected by the agent using tools for observation of the external world. The agent associates each observation (and its result, i.e. determination of the properties assigned or not to the objects) with a specific point in time t. Knowledge of this type is empirical in nature and is given as a set of all the observations made. The results of observations are stored in specialized databases, encapsulated (enclosed) in the agent's "body".

In the BGM model, the agent generates fixed language formulas, which are formal representations of the state of knowledge assumed by the agent about the objects $o_1, o_2, ..., o_M$ described by the properties $P_1, P_2, ..., P_K$. The simplest non-modal atomic formulas (statements) have the form:

- $p_i(o_j)$ has an intuitive meaning "Object o_j exhibits property P_i",
- $\neg p_i(o_j)$ has an intuitive meaning "Object o_j does not exhibit property P_i".

Unary predicate symbols p_1, p_2, ..., p_K are related to properties P_1, P_2, ..., P_K respectively. Whereas epistemic modality is expressed by the use of operators that allow the creation of modal atomic formulas on the basis of a non-modal formula φ:

- $Pos(\varphi)$ has an intuitive meaning "It is possible that φ",

- $Bel(\varphi)$ has an intuitive meaning "I believe that φ",
- $Know(\varphi)$ has an intuitive meaning "I know that φ".

For example, the formula $Pos(\neg p_2(o_3))$ means "It is possible that object o_3 does not exhibit property P_2", and $Bel(Happy(John))$ means "I believe that object $John$ exhibits property $Happy$".

In the works [31, 32, 92] and others, more complicated cases of formulas, such as conjunctions, alternatives, etc. were also analyzed, but in this chapter we will limit the subject to atomic modalities.

In the BGM model, a specific modal formula is considered to be properly grounded in the agent's current state of knowledge if and only if a strictly defined set of conditions imposed on that state of knowledge is met. These conditions are defined by the so-called **epistemic satisfaction relation** of a formula (a modal statement), which function is to represent cognitive semantics. Conceptually, this relation corresponds to the edge of the semiotic triangle that connects the material sign of language with the agent – as shown in the figure 3.5. The BGM model does not distinguish between the physical entity and the object representing it in the agent structures. Therefore, it is a simpler model than the one presented in this dissertation.

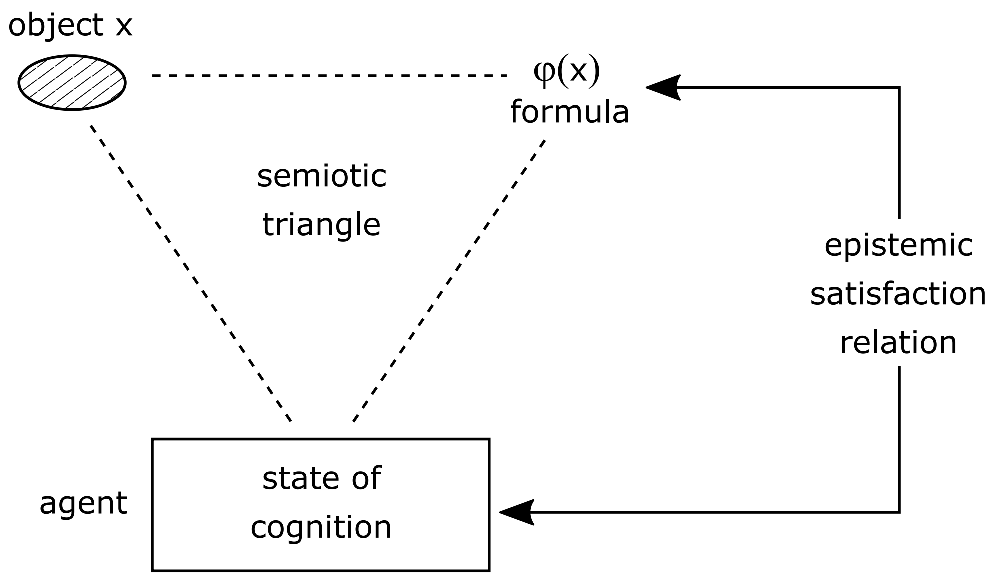


Figure 3.5: Epistemic satisfaction relation.

In the case when the agent observes the occurrence or absence of the property P in the object o at a given moment, the formula expressing certainty can be grounded, and therefore provided with the $Know$ operator. If the agent is unable to determine the current state of the property, it must refer to empirical knowledge. The more numerous empirical experiences of the agent in the presence of the property P in the object o, the stronger the agent's confidence about the possibility of the presence of the property P in the object o at the moment. More formally, it is expressed by means of grounding sets, collecting the agent's experience confirming the occurrence or absence of a given property in o in the preceding moments of time. In turn, the inductive strength of the grounding sets is defined as the classically understood cardinality of the grounding sets, and is used to determine the relative grounding strength $\lambda(t, p_i(o))$ and $\lambda(t, \neg p_i(o))$ of atomic formulas in time point t. The relative grounding strength is therefore a measure that is determined statistically.

The concept of modality thresholds is another element of the BGM model. They allow to characterize the propositional attitudes adopted by the agent. The modality thresholds are a strictly mental category. They are developed by the natural agent as subjectively experienced boundaries that define the scope of the agent's appeal to modal operators of possibilities and beliefs. Establishing threshold values is undoubtedly one of the outcomes of the processes of semiosis. In BGM it is assumed that modality thresholds are closely related to the concept of relative grounding strength and the agent refers to them when deciding on use of the operators of possibilities and beliefs. There are two modality thresholds λ_{minPos} and λ_{maxPos} related to experiencing possibilities and two modality thresholds λ_{minBel} and λ_{maxBel} related to beliefs. The threshold λ_{minPos} is to represent the subjectively perceived intensity of experiences, the occurrence of which (for a specific area of remembered empirical experiences) is to determine the agent's readiness to use the Pos operator. In turn, the threshold λ_{maxPos} determines the upper limit of the intensity of the remembered empirical experience, allowing the use of the possibility operator. Exceeding this threshold means that the linguistic representation of the realized propositional attitude requires referring to another modal operator. Modality thresholds λ_{minBel} and λ_{maxBel} are used in analogous way for Bel operator.

By comparing the relative grounding strengths $\lambda(t, p_i(o))$ and $\lambda(t, \neg p_i(o))$ with the set of the above thresholds, the agent determines whether there is a relation of epistemic satisfaction of individual modal atomic formulas. Formulas for which this relationship takes place are considered well-established and can be communicated by the agent to the outside world.

In the above-mentioned works, the properties of the BGM model were analyzed and the conditions to be met by the modality threshold system were defined, in order for the grounded formulas to be consistent with the desired linguistic behavior of the agent. In particular:

- the agent must be systematically unable to simultaneously ground formulas which, due to their intuitive content, never coexist in the description of the same state of knowledge; for example, the grounding mechanism cannot allow the simultaneous grounding of two modal formulas $Know(p(o))$ and $Bel(p(o))$,

- the agent must be equipped with a mechanism obligatorily grounding the collections of formulas which, due to their intuitive understanding, coexist with each other in the description of particular state of knowledge,
- the agent must demonstrate flexibility of behavior in terms of the simultaneous grounding of these formulas, which due to their intuitive meaning may or may not be grounded in the same state of knowledge.

More precisely the BGM model will be presented in the chapter 5.6 where, among others, formal definitions of the above concepts will be provided. In addition, the relationship between the model proposed in this dissertation and BGM will be presented. The signaled relationship between the models will be used to analyze the properties of the model originally proposed in this work. This will lead to proving for this model the properties analogous to the BGM model.

3.4 Conceptual spaces

Gärdenfors proposes conceptual spaces as a framework for modeling the formation and the evolution of concepts[15, 16]. These are geometrical structures based on quality dimensions and distance measures. Together with notion of prototypes taken from Rosch theories, they allow for Voronoi tessellation of conceptual space between categories. Gärdenfors states[17] that "(...) a conceptual level of representation should play a central role in the cognitives sciences. After having been dominant for many years, the symbolic approach was challenged by connectionism (which is nowadays broadened to a wider study of dynamical systems). However, for many purposes the symbolic level of representation is too coarse, and the connectionist too fine-grained. In relation to the two goals of cognitive science, I submit that the conceptual level will add significantly to our explanatory capacities when it comes to understanding cognitive processes, in particular those connected with concept formation and language understanding."

Conceptual spaces are technical model that allows for building and managing categories. It also uses so-called Region Connection Calculus to reason about categories. The models presented in this dissertation are also technically feasible and refer to a similar concept: space of objects in the sense that space is a set of objects additionally equipped with an appropriate numerical measure, e.g. distance or similarity of these objects. Moreover, both models use the concept of a prototype as a selected object from this space. However, the models in this work do not require boundaries between categories. Furthermore, the dissertation contains original cognitive semantics that enhances the explanatory power of the model.

3.5 Conclusions

The presented review shows several important premises for the research tasks defined in the chapter 1.2. Effective implementation of the phenomenon of grounding statements in ontological structures representing conceptual categories with a prototype, requires to develop a system architecture on a general level, including the following components:

- the model of category representation that includes the representation of object, the prototype of category, and the measure of the similarity or the distance of objects to the prototype,
- the category learning and updating mechanism (at this dissertation, the focus will be on learning with the teacher),
- the mechanism of grounding modal atomic statements in a knowledge system with an ontology based on conceptual categories with a prototype.

These elements must be implemented regardless of the methodology of knowledge representation adopted for a given artificial system.

Chapter 4

Model of the category learning strategy with a prototype

4.1 Description of the strategy

4.1.1 Data acquisition

We assume that the agent observes the environment using its sensors and that it is able to identify the entities in its surroundings. In the case of the physical world, it can do so based on the physical position of entities. Thanks to its sensors, the agent is able to assign values to the features (attributes) that describe the entity. The set of attributes describing the entity is constant and depends on the field of application and the agent's perception capabilities. The entity is represented in the agent's working memory by an object that is a combination of the values of the above attributes. The period of time in which the agent's environment does not change – that is, it consists of the same objects with the same values of attributes – is called an episode and treated as a single moment of time.

As already mentioned, learning will take place with the participation of the teacher, so this is a case of supervised learning. The aim of the learning process is to map the concepts used by the teacher in the agent's internal cognitive structures, to the extent that it is possible to generate statements regarding the membership of objects to the category. Let us note that the cognitive structures of the agent (artificial system) and the teacher (living or artificial system) may have a completely different architecture. Nevertheless, we expect the agent to be able to learn the concepts well enough to be able to generate statements that the teacher deems rational. Since the agent's cognitive structures in this model are based on the categorization theory developed as a part of work on human psychology, the expectation is justified that the statements will be rational from a human point of view.

The teacher can be a single entity or set of entities interacting with the agent. In the latter case, additional uncertainty may arise in the agent, due to the fact that individual teachers may have different conceptual systems. This may be caused by the following reasons:

- each teacher may have (and usually has) different experiences, that is, they have experienced a different set of episodes (situations, moments in time),
- even if two teachers experience the same situation, their perception of the world may be different, e.g. depending on the physical situation of the teacher, they may or may not have access to certain physical features of the object,
- each teacher may have differently constructed cognitive structures,
- each teacher's perception of the world may be different, e.g. one teacher may have a color camera, while the other only monochrome one.

All of the above issues can also be applied to the differences between teachers and the agent.

The greater the difference in teachers' perception of the world, the greater the agent's uncertainty about the application of a learned category system. To get rid of this aspect of uncertainty, it must be ensured that all teachers assign given objects to categories in exactly the same way. In the case of agents acting in a real environment, the concept system used by teachers is socially sanctioned, so quite similar for all teachers, although rather not identical. Of course, the problem disappears when there is only one teacher.

Later in the work we will use the singular term "teacher", although in general there may be more teachers. However, this does not change the model presented below.

We assume that the teacher is able to name objects in a certain area of environment by assigning labels to them, which are signs referring to categories from the system of concepts used by the teacher. We also assume that there is a mechanism for assigning a label to an object, available to the agent. In this way, the agent receives a label-object association, whereby one object may be assigned multiple labels and many objects may be assigned to the same label. In this paper, labels will be in the form of character strings, where we distinguish two types:

- **positive labels** – assigned to an object by the teacher if they believe that the object belongs to the category indicated by the label,
- **negative labels** – assigned to an object by the teacher if they believe that the object does not belong to the category indicated by the label.

For the sake of clarity of the description, in examples presented in this chapter, we will assume that when the learning process begins, the cognitive structures are empty, i.e. they do not contain any information. However, since the agent model presented here contains an explicit knowledge representation, it can be equipped with some initial knowledge in the form of ready-made structures.

4.1.2 Updating of cognitive structures

Each appearance of the object-label linked pair causes an update of the cognitive structure describing the category indicated by the label. This structure will be called **the cognitive model of category**. In this work it is proposed that the cognitive model of the category (concept) c includes references to all cases of observations (experiences) of objects that were labeled by the teacher with a positive or negative label indicating the category c. However, the agent will not remember the references to a specific object, but rather to its counterpart in mental space. Thus, the observation of the object x in the episode referring to the time point t stored in the model of category c includes:

- information on the value of the x object features observed by the agent in the episode referring to the time point t,
- information on the labels given by the teacher to the object x in the episode referring to the time point t.

Let us note, that the cognitive model does not contain the following information regarding the above observation:

- the identifier that physically distinguishes the x object from other objects,
- the identifier distinguishing the source of the labels (teachers),
- the reference to the episode in which the observation occurred.

The set of labels associated with a given object may be empty. Such experience will not be part of any of the agent's cognitive models, but the object may be described by the agent using statements about the object's membership to the category.

The prototype is determined on the basis of all the observations stored in a given model. According to psycholinguistic theories, it should be the most representative member of the category. This does not mean that it must be a real observation of the entity, it can be an abstract object, i.e. such a combination of feature values that never appeared in the agent's experience. The prototype should contain as many as possible of the most important features of the objects belonging to the category. How to determine which features are in fact the most important and how to choose the prototype will be the subject of research in this dissertation.

Designating a prototype enables its use to organize the structure of the category model. For this purpose, apart from the prototype, it is necessary to define a computable mechanism allowing for the comparison of two objects. We call this mechanism **the macrostructure** and define it as a computable function, which is e.g. a measure of the mental distance between two objects or the similarity of two objects. In turn, the microstructure determines the internal structure of the object, it can generally be a set of attributes, a tree structure, etc.

The macrostructure allows to sort the objects belonging to the model according to the value of the above measure in relation to the prototype. In this way, we obtain the prototypical effect of the agent's "feeling" that certain objects are better examples of a given category and others are worse. According to the results of the research described in the chapter 3.2, this effect may lead to the division of objects in mental space into three areas:

- objects certainly belonging to the category – we call this area **the core of the category**,
- objects certainly not belonging to the category – we call this area **the outer region of the category**,
- questionable objects that may belong to a category, but it is not certain – we call this area **the boundary of the category**.

This model assumes a radial category structure with the prototype as the central element. Schematically, the model areas can be shown as in the figure 4.1, where the point $o_c^\star$ symbolizes the prototype of the category c. If we define a macrostructure as a measure of the distance from the prototype, then the distance of the point from the prototype in the picture is proportional to the value of this measure (the closer it is, the smaller the distance). If we define the macrostructure as a measure of similarity to the prototype, then the distance of the point from the prototype in the picture is inversely proportional to the value of this measure (the closer, the greater the similarity).

By defining the macrostructure, it is also possible to determine the values of τ_c radii, which allow to unambiguously describe all three areas. As one can see, τ_c^+ is the extent of the core of the category, while τ_c^- is the extent of the boundary of the category. We assume that for any model of category in which both radii are defined, the following condition holds:

$$\tau_c^+ < \tau_c^-. \tag{4.1.1}$$

The above condition seems to be reasonable in common sense, but it will also prove necessary for the proper linguistic behavior of the agent, as will be shown in the chapter 5.5.

The smaller the core and the larger the boundary in the cognitive model of the category, the greater the uncertainty with which the agent can assign objects to this

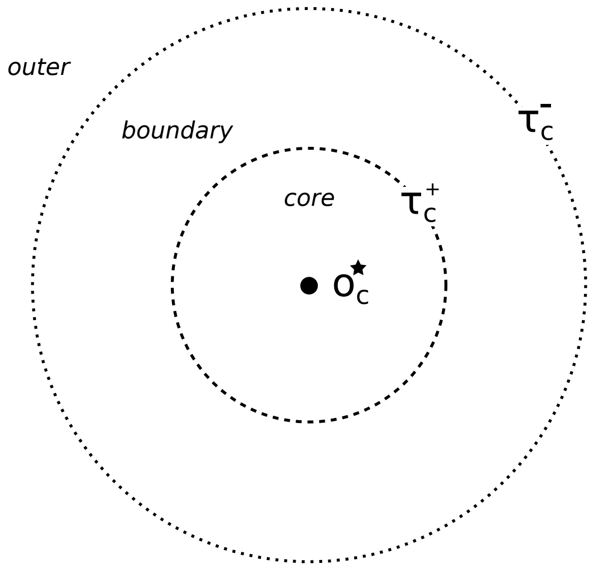


Figure 4.1: Schematic picture of three areas of the model of a category.

category. In the extreme case, the core of the category may be empty, which means that the agent is unable to say anything certain about whether the objects belong to the category. The reasons for this may be as follows:

- the proposed structure of the cognitive model with a prototype is not suitable for the representation of the category – e.g. the similarity of the category elements to the prototype is as low as the non-category elements,
- the set of features available to the agent's perception is insufficient to distinguish between observations belonging to a category and not belonging to a category,
- the granularity of the values of the features available to the agent's perception is insufficient to distinguish between observations belonging to the category and not belonging to the category,
- the values of the features available to the agent's perception are burdened with too many erroneous readings,
- the agent received too little label information, or even received incorrect label-to-object assignments,
- the agent observed too few examples to correctly determine the prototype and/or areas of the category model (i.e. has too little experience),
- if there is more than one teacher – too large differences in the assignment of objects to categories by different teachers.

Based on the sizes of the core and the boundary of category, specific conditions or benchmarks can be defined that allow to assess the quality of the model found. For example, the first step in determining the prototype should be to find a set of prototype candidates. For this purpose, it is necessary to define the quality conditions that each candidate must meet. The prototype can be selected randomly from the above set or an additional measure can be defined that will allow the selection of the best candidate for the prototype. Generally we expect that it would be the object that allows to obtain the largest category core, with the smallest possible boundary.

Recognizing when the model is likely to fail is crucial for wide-scale adoption in safety critical domains such as robotics and control[37]. For example, in neural networks there exist several approaches for estimating epistemic uncertainty which are pretty complicated or time-consuming [1, 50]. An important feature of the proposed model is the ability for the agent to detect categories that it is not able to model well. These will be the categories in which the core and/or the boundary do not meet the specified qualitative conditions. We will call these category models **ill-defined models**. Category models meeting the above conditions will be called **well-defined models** by analogy. When the agent detects that the model of a certain category is ill-defined, it can communicate this to the teacher. The agent will not, however, generate statements about the membership of objects to the category with ill-defined model. In this way, we should obtain greater rationality and certainty of the agent's operation.

Algorithm 4.1 presents in general the above ideas about updating the agent's cognitive structures.

4.2 Agent system model

4.2.1 External world's model

The agent's perception enables the observation of atomic objects in some area of its environment. These observations are grouped into episodes that are connected to relativized points in time (snapshots) $t \in T = \{t_0, t_1, t_2, ...\}$. A linear order $\leqslant^{TM}$ is defined above the set T.

By A we will denote a set of attributes representing the features of objects. V_a will be the set of values that the attribute $a \in A$ can take. We assume that the sets A and V_a are constant for all episodes experienced by the agent. The domain of the attribute depends on the use of the agent, hence it can include sets of values:

- binary – e.g. $\{0, 1\}$, $\{yes, no\}$,
- nominal – e.g. $\{large, medium, small\}$, $\{red, green, blue, black\}$,
- numeric – e.g. set of natural numbers, set of real numbers.

Algorithm 4.1: A general outline of the strategy for updating of cognitive model of category.

Input: cognitive model m_c of the category c,
learning set consisting of observations related to c.
Output: updated cognitive model m_c.

pre-process the learning experience;
determine a set *Candidates* consisting of chosen prototype candidates;
while *there are still Candidates and a suitable prototype* $o_c^\star$ *has not been found* **do**
    pick (and remove) an object o from *Candidates*;
    evaluate *Core* and *Boundary* determined by o, basing on macrostructure;
    calculate quality parameters related to *Core* and *Boundary*;
    if *Core and Boundary are acceptable* **then**
        select o as a prototype, assign $o_c^\star := o$;
        set the prototype $o_c^\star$ and radii τ_c^+, τ_c^- in the model m_c;
        the model m_c is well-defined;
if *no prototype has been found* **then**
    the model m_c is ill-defined;

In this dissertation, we will focus on the first two of the above types, but it does not limit the application of the presented model only to binary and nominal attributes.

The second perceptual ability of the agent is to receive labels assigned by the teacher to objects in the environment. The set of labels L contains positive and negative labels indicating categories. We will denote the labels symbolically as follows:

- *is-c* is a positive label indicating the category c, which means that the teacher believes that the object belongs to the category c,
- *not-c* is a negative label indicating the category c, meaning that the teacher believes the object does not belong to the category c.

Definition 4.1. The episode, i.e. the internal model of the state of the agent's environment, assigned by the agent to time point t is a tuple of the form $Episode(t) = \langle X_t, A, V, L_t, percept, label\rangle$, where

- X_t – a finite set of objects,
- A – a finite set of attributes,
- $V = \bigcup_{a \in A} V_a$, V_a – domain of attribute a,
- L_t – a finite set of labels,
- *percept* – a function representing observed values of attributes of objects, $X_t \times A \longrightarrow V$, such that $percept(x, a) \in V_a$ for all $x \in X_t$ and $a \in A$,

- *label* – a function representing the observed assignments of labels to objects, $X_t \longrightarrow \Pi(L)$, $\Pi(L)$ is a power set of L.

The above definition is therefore an extension of the classic information system[64] with a set of labels and the *label* function.

Each object in X_t receives an identifier that allows it to be associated with labels. The identifier is also necessary to relate the statements generated by the agent to the correct object. This problem is explained in more detail in chapter number 5, describing the cognitive semantics of the agent. We do not require the agent to be able to associate the same entity with the same object (and thus identifier) in different episodes. To emphasize this, we will use unique identifier for each object in each episode.

The *percept* function is a formal representation of the agent's perception of the environment. It is a total function, so it is defined for all values of x and a.

The *label* function reflects the mechanism of assigning a label to an object. The set of labels assigned to a given object may be empty, which means that the teacher has not named the object in any way.

Definition 4.2. At time point $t_n \in T$ the state of empirical knowledge about the external world is defined by a temporal collection of episodes given as follows:

$$Episodes(t_n) = \{Episode(t_i) : t_i \in T \quad and \quad t_i \leqslant^{TM} t_n\}.$$

Example 1. The table 4.1 shows an example of the agent's experience collected in two episodes, i.e. $Episodes(t_2) = \{Episode(t_1), Episode(t_2)\}$. The set A consists of three binary attributes, the meaning of which is as follows: a_1 – having a beak or bill, a_2 – having wings, a_3 – domesticated. In the first episode, the collection of objects $X_{t_1} = \{x_{1,1}, x_{1,2}, x_{1,3}\}$, while in the second $X_{t_2} = \{x_{2,1}, x_{2,2}\}$. In the first episode, the set of labels $L_{t_1} = \{$*is-bird*$\}$, while in the second $L_{t_2} = \{$*not-bird, is-mammal*$\}$.

episode	**object**	a_1	a_2	a_3	**labels**
$Episode(t_1)$	$x_{1,1}$	*yes*	*yes*	*no*	*is-bird*
	$x_{1,2}$	*yes*	*yes*	*yes*	*is-bird*
	$x_{1,3}$	*yes*	*yes*	*yes*	*is-bird*
$Episode(t_2)$	$x_{2,1}$	*yes*	*yes*	*no*	
	$x_{2,2}$	*no*	*no*	*yes*	*not-bird, is-mammal*

Table 4.1: Two exemplary episodes.

The presented table corresponds to the situation when in the first episode the agent observed three objects. All of them were marked by the teacher with a positive label *is-bird*, expressing their belief that the objects $x_{1,1}$, $x_{1,2}$ and $x_{1,3}$ fall into the category *bird*. In the second episode, there are only two objects in agent's range of perception.

The agent does not track objects between episodes, so it is unknown whether they are the same entities as in $Episode(t_1)$. The teacher marked the object $x_{2,2}$ with two labels: negative *not-bird* and positive *is-mammal*. This means that the teacher believes that the object $x_{2,2}$ does not fall into the category *bird*, but does belong to the category *mammal*. There are no labels associated with the object $x_{2,1}$.

4.2.2 Basic concepts and preprocessing of learning experience

The models of categories are not built on the basis of references to specific physical objects, but on the basis of mental images representing possible combinations of the values of attributes of the objects. For this reason, the image of the currently observed object in the working memory must be related to mental space in embodied ontology. Such a mental operation will be called preprocessing. For that purpose the universe of mental representations is defined, which is a Cartesian product of all attributes domains V_a. It constitutes a domain over which categories models including their prototypes are to be defined.

Definition 4.3. Universe of mental representations of distinguishable objects O is defined as:

$$O = \{o_1, ..., o_M\} \stackrel{\text{def}}{=} \bigtimes_{a \in A} V_a.$$

Other symbols and notions that will be used:

- $\Pi(O)$ – a power set of the universe O – it serves in strategies as a domain for sets of prototype candidates,
- $\hat{\Pi}(O)$ – a set of all multisets defined over the universe O – its members serve as a quantitative reflections of a cumulative experience related to a certain phenomenon (e.g., objects being confirmed as c),
- $(o, +)^c$ – a mental representation of a piece of a positive learning experience, that is, of a situation in which an object from an external world which has been perceived by the agent and reflected in its mental space as o, has been confirmed as belonging to category c,
- $(o, -)^c$ – a mental representation of a piece of a negative learning experience, that is, of a situation in which an object from an external world which has been perceived by the agent and reflected in its mental space as o, has been rejected from belonging to category c,
- $T_c = \{(o, +)^c : o \in O\} \cup \{(o, -)^c : o \in O\}$ – a universe of all possible outcomes of a set of learning episodes,

- $\hat{\Pi}(T_c)$ – a set of all multisets defined over T_c – a universe of multisets quantitatively aggregating an overall learning experience related to concept c,
- $Exp_c \in \hat{\Pi}(T_c)$ – a multiset quantitatively aggregating an overall learning experience related to a concept c – it serves as the learning set as it constitutes an input of a strategy for determining model of category c including its potential prototype.

An abbreviated notational convention for multisets will be used, that is, for the universe $\{a, b, c, d, e\}$ the multiset $\{a, d, b, a, b, b\} = \{2a, 3b, 0c, 1d, 0e\} = \{2a, 3b, 1d\}$, therefore in forthcoming chapters multisets $X \in \hat{\Pi}(O)$ are denoted as:

$$X = \{k_{o_1}o_1, k_{o_2}o_2, \ldots, k_{o_M}o_M\}, \qquad k_{o_1}, \ldots, k_{o_M} \in \mathbb{N} \cup \{0\}$$

where particular k_{o_i} denote multiplicities of respective objects belonging to X.

An analogous abbreviated representation is given for multisets $I \in \hat{\Pi}(T_c)$:

$$\begin{aligned} I =& \{k^+_{o_1}(o_1, +)^c, \ldots, k^+_{o_M}(o_M, +)^c, k^-_{o_1}(o_1, -)^c, \ldots, k^-_{o_M}(o_M, -)^c\}, \\ & k^+_{o_1}, k^-_{o_1}, \ldots, k^+_{o_M}, k^-_{o_M} \in \mathbb{N} \cup \{0\} \end{aligned}$$

where particular $k^+_{o_i}$ ($k^-_{o_i}$, respectively) denote multiplicities of learning episodes in which an object perceived as o_i was confirmed (rejected) as belonging to category c. In case of very large multisets, elements with multiplicity ($k^+_{o_i}$ or $k^-_{o_i}$) equal to 0 will be ommited.

Because the cognitive models of the categories will be based on multisets $Exp_c \in \hat{\Pi}(T_c)$, the agent must preprocess data collected from the environment. As a result of the above procedure, we obtain a set of Exp_c multisets, one for each model of category c that is in the agent's cognitive structures. The above categories can be given a priori or added to cognitive structures as they are indicated by the teacher with a positive or negative label.

Example 2. Continuing the example from the table 4.1, the following sets of labels appeared in subsequent episodes: $L_{t_1} = \{$*is-bird*$\}$ i $L_{t_2} = \{$*not-bird, is-mammal*$\}$. As one can see, they concern two categories: *bird* and *mammal*. The domain of each of the attributes $a_i \in A$ is also binary: $\{yes, no\}$. For this reason $|O| = 2^3$, and the space of mental representations consists of the following objects:

$$\begin{array}{llll} o_1 = [no, no, no] & o_2 = [no, no, yes] & o_3 = [no, yes, no] & o_4 = [no, yes, yes] \\ o_5 = [yes, no, no] & o_6 = [yes, no, yes] & o_7 = [yes, yes, no] & o_8 = [yes, yes, yes]. \end{array}$$

The question at this point is how many of the episodes available to the agent should enter its cognitive structures. It is easiest to assume that all available episodes are reflected in them, but one can also limit their number, e.g. to n last episodes, which would result in the agent "forgetting" some information. Assuming that in this example both episodes have been processed into the agent's cognitive structures, we obtain the following multisets:

$$\begin{aligned} Exp_{bird} = \{&0(o_1,+)^{bird}, 0(o_2,+)^{bird}, 0(o_3,+)^{bird}, 0(o_4,+)^{bird},\\ &0(o_5,+)^{bird}, 0(o_6,+)^{bird}, 1(o_7,+)^{bird}, 2(o_8,+)^{bird},\\ &0(o_1,-)^{bird}, 1(o_2,-)^{bird}, 0(o_3,-)^{bird}, 0(o_4,-)^{bird},\\ &0(o_5,-)^{bird}, 0(o_6,-)^{bird}, 0(o_7,-)^{bird}, 0(o_8,-)^{bird}\} \end{aligned}$$

$$\begin{aligned} Exp_{mammal} = \{&0(o_1,+)^{mammal}, 1(o_2,+)^{mammal}, 0(o_3,+)^{mammal}, 0(o_4,+)^{mammal},\\ &0(o_5,+)^{mammal}, 0(o_6,+)^{mammal}, 0(o_7,+)^{mammal}, 0(o_8,+)^{mammal},\\ &0(o_1,-)^{mammal}, 0(o_2,-)^{mammal}, 0(o_3,-)^{mammal}, 0(o_4,-)^{mammal},\\ &0(o_5,-)^{mammal}, 0(o_6,-)^{mammal}, 0(o_7,-)^{mammal}, 0(o_8,-)^{mammal}\} \end{aligned}$$

It should be emphasized here, that there is a difference between the objects from the sets X_t and O. The former are representations of objects present in the agent's environment and its available perceptions. The latter are mental representations of all objects that can be "imagined" by an agent. The difference between these sets and the course of preprocessing are shown in the figure 4.2. It shows how objects $x_{i,j}$ observed by agent in example 1 are related to objects o_k in mental space.

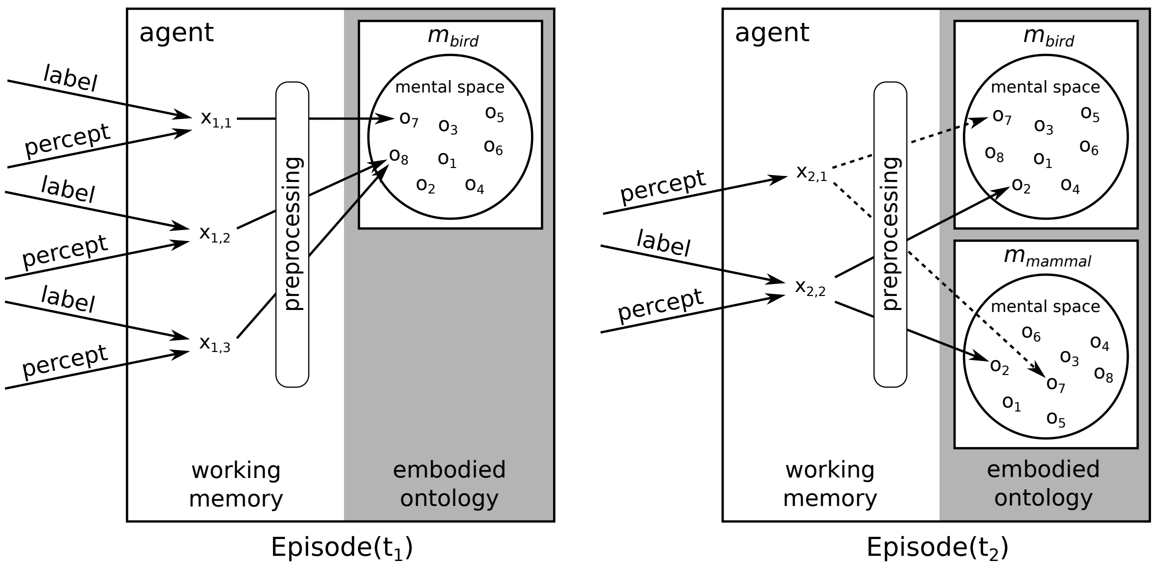


Figure 4.2: Preprocessing of two exemplary episodes.

The figure on the left shows objects from $Episode(t_1)$. The object $x_{1,1}$ is observed realization of mental object o_7. The objects $x_{1,2}$ and $x_{1,3}$ are both observed realizations of one mental object o_8 – the above objects have the same combination of values of attributes. The agent differentiates objects $x_{1,2}$ and $x_{1,3}$ basing on their physical location. Since objects $x_{1,j}$ are all marked with labels referring to category *bird*, they are used to update mental structures of model of category *bird* – denoted m_{bird}. A set of all models m_c will be denoted as M. The models in M can be given a priori or constructed as needed, e.g. after first encounter of label describing new category.

The figure on the right shows objects from $Episode(t_2)$. The object $x_{2,1}$ is observed realization of mental object o_7. It is not marked with any label by the teacher, so it is not going to be used to modify structure of any category's model. Still, it can be preprocessed if other mental procedures are to be executed, like for example grounding of statement regarding object $x_{2,1}$. The object $x_{2,2}$ is observed realization of mental object o_2. It is marked with two labels – referring to categories *bird* and *mammal*. Thus, object $x_{2,2}$ is used to update mental structures of models m_{bird} and m_{mammal}. Let us note, that all models are constructed basing on the same universe of mental representations of distinguishable objects O, but in each model this universe can be structured differently. It is shown in the figure 4.2, as objects o_i are placed differently in models m_{bird} and m_{mammal}. The structure imposed on universe O will be described in detail in the next chapters.

Note that in the agent's cognitive structures there is no information on the basis of which episodes and objects in the agent's working memory the data contained in the Exp_c multisets were obtained. This information is lost during preprocessing. For example, the multiplicity of the element $(o_2, +)^{mammal}$ in multiset Exp_{mammal} is equal to 1 on the basis of data obtained about the object $x_{2,2}$ in $Episode(t_2)$, but this fact will not be stored in memory. The further stages of processing the agent's learning strategy will be the same, regardless of on which objects $x_{i,j}$ and in which episodes the multiplicity of the element $(o_2, +)^{mammal}$ was determined.

The multiset Exp_c can be obtained in two ways:

- calculated when necessary for a given category, based on the $Episodes(t)$ base of experience, and then deleted – as a result, the memory requirements are small because we process a multiset for only one cognitive model as part of the process related to a given category; however, it is more computationally demanding,
- calculated and stored for all models and updated after each episode – this reduces the number of calculations but increases the agent's memory requirements.

4.3 Cognitive model

The agent's cognitive structures consist of cognitive models, with one cognitive model corresponding to each category.

Definition 4.4. Cognitive model m_c of category (concept) c consists of:

- Exp_c – described above multiset quantitatively aggregating an overall learning experience related to a concept c,
- prototype $o_c^{\star} \in O$,
- threshold value (radius) τ_c^{+} separating the core of a category from the boundary of category,
- threshold value (radius) τ_c^{-} separating the boundary of category from the outer region of category.

Additionally, in order to be able to determine the cognitive model, it is necessary to define:

- a macrostructure being a function $f : O \times O \to \mathbb{R}^{+} \cup \{0\}$,
- a mechanism for determining the set of candidates for a prototype, which is a function $extractCandidates : \hat{\Pi}(O) \to \Pi(O)$.

Macrostructure f allows for a computational skill to apply a particular measure of distance or similarity (proximity) to evaluate and numerically express the level of mentally experienced difference of mental representations of two objects. Multiple computational models to measure distance and similarity of ontological structures were developed which seem appropriate to support the autonomous and encapsulated development of categories with prototypes by artificial agents, e.g. [47]. In practical contexts their values are usually computed basing on internal organization of objects being compared and the internal organization is sometimes called the microstructure of cognitive universe.

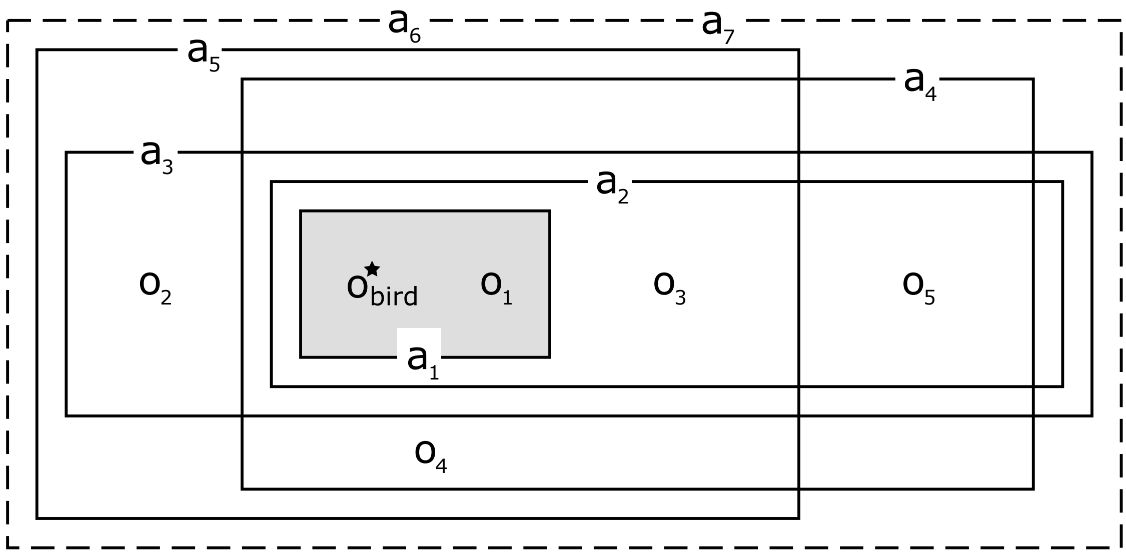
The ability to compare mental representations of objects and numerically express the result of such comparison has a direct application to multiple steps of the strategy, e.g. while the intensity of relationship of objects with a potential or actual prototype of a category is determined. It can also be additionally emphasized that the nature of comparison cognitive tool used in a specific practical context (i.e. distance vs. similarity) has a significant impact on the content of decisions controlling the flow of strategy within a specific implementation of the proposed strategy. However, it does not change its overall logic.

The second function $extractCandidates$ allows for a computational skill to determine mental representations of objects which can be treated (perceived) by artificial cognitive agents as representations of mental centers for collections of objects. For each $X \in \hat{\Pi}(O)$ object $o \in extractCandidates(X)$ if it meets the definition of a central object, called **centroid**, formulated for a specific practical context. The nature of central objects can vary from one application to another. However, in the majority of practical situations they are required to fulfill a particular optimization criterium, formally and pragmatically consistent with the nature of cognitive universe O and the internal nature of its objects.

It is also possible to impose additional requirement on central objects, namely the following inclusion $extractCandidates(X) \subseteq X$, that is the expectation that central objects are **medoids** of the set X, rather then its centroids, see [91]. Again, it is worth mentioning that the issue just raised is not purely theoretical, just as it is not irrelevant to particular contexts whether the prototype for a collection of observed objects is to never be observed (i.e. centroid), and not one of the objects observed at least once (i.e. medoid). For example, in chapter 7 the model is tested on a few datasets. One of them is based on 1984 United States congressional voting database, that contains democrat and republican congressmen votes on various topics. From that data it is possible to derive models of *democrat* category and *republican* category. In these models it is possible to find the prototypical democrat and republican congressmen. They don't have to represent actual people, as it is not important for political analysis, so prototypes can be centroids. On the other hand, in chapter 7 there is also a dataset containing information about patients with different types of tumors. It seems that in that case, a prototypical patient that has e.g. lung tumor should represent the actual person, as there might be connections between attribute values that are important for medical analysis.

In the above model, the category has an internal prototype structure and allows obtaining effects corresponding to the theses of the standard version of the prototype semantics, presented in the chapter 3.2, i.e. that the degree of representativeness of a given instance corresponds to the degree of its belonging to the category, elements of a given category do not have to have properties common to all elements, the boundaries of the category are fuzzy, and the belonging to a given category is based on the degree of similarity to the prototype. The connections to the theses of the standard version of the prototype semantics is analyzed in more detail in the chapter 4.5.

If we take the distance between objects as the macrostructure f, then for the object o the degree of membership and representativeness for a given category c corresponds (inversely) to the value of the measure of distance from the prototype, i.e. $f(o, o^{\star}_{c})$. Thanks to this, elements of a given category do not have to have features (values of attributes) common to all elements – they are connected by similarity to the prototype (the greater the distance, the smaller the similarity). As in the examples from the chapter 3.2, category elements (objects in mental space) can be ordered according to the measure of distance from the prototype. An example of a situation is shown in the figure 4.3, which corresponds to the figure 3.2, showing the relationship between the elements of the *bird* category.



$f(o^\star_{bird}, o^\star_{bird}) = 0 \quad f(o_1, o^\star_{bird}) = 0 \quad f(o_2, o^\star_{bird}) = 3$
$f(o_3, o^\star_{bird}) = 1 \quad f(o_4, o^\star_{bird}) = 3 \quad f(o_5, o^\star_{bird}) = 2$

Figure 4.3: Exemplary cognitive model of category *bird* with prototype.

Below the figure there are examples of the measure of the distance from the prototype $o^\star_{bird}$. As one can see, we expect that the objects with the most features in common with the prototype, also have the smallest distance value. The o_1 object even has a value of 0, same as $o^\star_{bird}$, because it has exactly the same attribute values as the prototype. As it was already mentioned, such an object does not always have to be present. The ordering of the objects according to the increasing distance from the prototype (i.e. decreasing similarity) will look like this: o_1, o_3, o_5, $\{o_2, o_4\}$.

4.4 Definition of the strategy

4.4.1 Algorithm

The algorithm 4.2 presents in more detail the strategy of learning of the category, or more precisely, of updating the cognitive structures of the agent. Its notation assumes that the macrostructure is some measure of distance. If a measure of similarity were to be used, the logic of some conditions and operations would have to be reversed.

At the input of the algorithm there is a set of episodes, as already mentioned it may cover all or part of the agent's experiences. The algorithm works on the cognitive model m_c representing a given category, updating it. The m_c model in the algorithm input may be empty. The algorithm should be executed for a given category when new information concerning this category becomes available. The result of the algorithm is

Algorithm 4.2: Prototype based strategy of learning categories.

Input: cognitive model m_c of the category c,
set of episodes $Episodes(t)$.
Output: updated cognitive model m_c.

$Exp_c := Preprocess(Episodes(t))$;
initialize a chosen prototype as $o_c^\star := NULL$;
$E^+ := E_c^+(Exp_c)$;
$E^- := E_c^-(Exp_c)$;
compute the set $Candidates := extractCandidates(\hat{E}_c^+(Exp_c))$;
while $Candidates \neq \varnothing \wedge o_c^\star = NULL$ **do**
    choose a prototype candidate $o \in Candidates$;
    $Candidates := Candidates \setminus \{o\}$;
    compute distance values $f(o^+, o)$ for $o^+ \in E^+$;
    compute distance values $f(o^-, o)$ for $o^- \in E^-$;
    $f^-_{\min}(o) := \min\limits_{o^- \in E^-} \{f(o^-, o)\}$;
    $f^+_{\max}(o) := \max\limits_{o^+ \in E^+} \{f(o^+, o)\}$;
    $F^+ := \{f(o^+, o) : o^+ \in E^+ \wedge f(o^+, o) < f^-_{\min}(o)\}$;
    compute a radius of the core $\tau_c^+ := \begin{cases} \max\{f \in F^+\} & F^+ \neq \varnothing \\ NULL & F^+ = \varnothing \end{cases}$;
    $F^- := \{f(o^-, o) : o^- \in E^- \wedge f(o^-, o) > f^+_{\max}(o)\}$;
    compute a radius of the boundary $\tau_c^- := \begin{cases} \min\{f \in F^-\} & F^- \neq \varnothing \\ NULL & F^- = \varnothing \end{cases}$;
    `// Compute a core of potential` c.
    if $\tau_c^+ \neq NULL$ **then**
        $Core_c(o) := \{o^+ : o^+ \in E^+ \wedge f(o^+, o) \leqslant \tau_c^+\}$
    else
        $Core_c(o) := \varnothing$;
    `// Compute an outer of potential` c.
    if $\tau_c^- \neq NULL$ **then**
        $Outer_c(o) := \{o^- : o^- \in E^- \wedge f(o^-, o) \geqslant \tau_c^-\}$
    else
        $Outer_c(o) := \varnothing$;
    `// Compute a boundary of potential` c.
    $Boundary_c(o) := (E^+ \cup E^-) \setminus (Core_c(o) \cup Outer_c(o))$;
    if $|Core_c(o)| \geqslant |Boundary_c(o) \cap E^+|$ **then**
        assign $o_c^\star := o$;
        add a category c with a prototype $o_c^\star$ and τ_c^+, τ_c^- to the ontological knowledge base of the agent;
if $o_c^\star = NULL$ **then**
    the model m_c is ill-defined and has not been learned;

an updated model m_c containing the prototype $o_c^\star$, radius τ_c^+ and radius τ_c^-, as long as the category is well defined. If the category is ill-defined, the above parameters cannot be determined.

In the first step of the algorithm, the set Exp_c is determined based on the set of episodes, as described in the chapter 4.2.2. Next steps (lines 2-4) define a variable $o_c^\star$ used to evaluate a STOP condition of the algorithm, determine sets of objects confirmed as c (the set $E_c^+(Exp_c)$) and rejected as c (the set $E_c^-(Exp_c)$) within the learning experience.

Definition 4.5. $E_c^+(Exp_c) = \{o : o \in O \wedge k_o^+ > 0\}$ where $Exp_c \in \hat{\Pi}(T_c)$.

It is a set of all mental representations of objects at least once confirmed as c in the learning experience Exp_c.

Definition 4.6. $E_c^-(Exp_c) = \{o : o \in O \wedge k_o^- > 0\}$ where $Exp_c \in \hat{\Pi}(T_c)$.

It is a set of all mental representations of objects at least once rejected as c in the learning experience Exp_c.

Definition 4.7. $\hat{E}_c^+(Exp_c) = \{k_o^+ o : k_o^+(o,+)^c \in Exp_c\}$ where $Exp_c \in \hat{\Pi}(T_c)$.

It is a multiset constituting a quantitative reflection of a cumulative experience gathered in Exp_c and related to objects being confirmed as c.

A testing set *Candidates* (line 5) is initialized using a procedure *extractCandidates* ($\hat{E}_c^+(Exp_c)$) which determines hypothetical candidates for a prototype. This step indirectly evaluates a multiset quantifying an overall learning experience (a multiset $\hat{E}_c^+(Exp_c)$ or possibly whole Exp_c multiset). Its result depends on chosen macrostructure and the conditions imposed on candidates, for example that they have to be centroids or medoids. Specific examples of implementations of this procedure are described in chapter 6. This procedure could involve elaborate analysis but it is advised to keep it less complicated and not too computationally intensive.

The whole procedure runs until it succeeds, that is, a category with a prototype $o_c^\star$ is formed, or until it fails by running out of candidates to evaluate. The presented procedure is non-deterministic in the sense that it starts with an arbitrary object o from the set of candidates (line 7). The non-determinity could be partially reduced by ordering the members of *Candidates* in a decreasing order according to the number of times they have been confirmed as c, and further, by always picking the most promising (in terms of the mentioned number) candidate from the set.

For a chosen o the procedure evaluates a distribution of objects (from the learning experience) within an agent's mental space in relation to o. It does this by directly calculating values of distance between o and particular objects from E^+ and E^-. It evaluates, for candidate o, a value $f^-_{\min}(o)$ (line 11) which becomes an upper limit which cannot be crossed by a chosen value of a radius of the *Core*. This limit guarantees that the *Core* contains only confirmed cases. A particular value of core's radius chosen in the algorithm is set to a distance to the farthest (from o) positive observation from E^+ but still closer than $f^-_{\min}(o)$ (lines 13-14). It could be modified according to other commonsense justifications, e.g. to set the radius of the core to reach until the closest negative example itself. In that case the radius would reach as far as possible until a counter-example is found.

Similarly, it evaluates, for candidate o, a value $f^+_{\max}(o)$ (also line 12) which becomes a lower limit which cannot be crossed by a chosen value of a radius of the *Boundary*. This limit guarantees that the *Outer* contains only rejected cases. A particular value of boundary's radius chosen in the algorithm is set to a distance to the closest (from o) negative observation from E^- but still further than $f^+_{\max}(o)$ (lines 15-16).

Objects from the learning experience are further divided into $Core_c(o)$, $Outer_c(o)$, and $Boundary_c(o)$ (lines 17-25) in order to analyze their properties and decide whether o becomes a successful candidate for the prototype. These steps can be seen as a preparation for the evaluation of quality parameters (line 6 of the meta-strategy presented in algorithm 4.1).

For $Core_c(o)$, $Outer_c(o)$ and $Boundary_c(o)$ there is an analogy in **rough set theory**[63], where the category core corresponds to the lower approximation of the set, the category boundary corresponds to the set boundary, and the outer region of the category corresponds to the complement of the upper approximation of the set. However, the concepts in rough set theory are defined based on the relation of indiscernibility[85] or dominance[21, 86, 101], while here we define them in relation to the macrostructure expressing distance or similarity to the prototype.

The evaluation condition is explicitly checked in the line 26. If the condition is satisfied, the candidate o gets accepted as a prototype (line 27) and together with resulting radius of the core (τ_c^+) and radius of the boundary (τ_c^-) it is added to the ontological knowledge base of the agent (line 28). If that happens, the procedure ends successfully since $o_c^\star$ has been chosen. Otherwise, if it fails for all candidates, it is considered that category can not be learned in this model of category (lines 29-30).

All objects o for which $f(o, o_c^\star) \leqslant \tau_c^+$ belong to the core of the category. All objects o for which $f(o, o_c^\star) \geqslant \tau_c^-$ belong to the outer region of the category. All objects o for which $\tau_c^+ < f(o, o_c^\star) < \tau_c^-$ belong to the boundary of the category. Calculation of the τ_c^+ and τ_c^- threshold values has the advantage that it allows to compare the distance of objects from the prototype with the above radii, instead of with other objects belonging to the category model. This reduces the computational complexity and enables the generalization of the accumulated knowledge, because on the basis of the above conditions also objects not existing in the model (e.g. not labeled with a label indicating a given category) can be assigned to one of the three areas of the model.

Still, the condition from line 26 calls for at least some explanation as it might look as if chosen rather arbitrarily. The acceptance condition for a well-formed category is, in general, very domain specific. The condition listed in the algorithm calls for at least half of the objects confirmed at least once as c in the learning experience, to be located within $Core_c(o)$. It is a very strict condition which would, for example, most probably fail in domains (or societies) in which the nature of concepts' meaning is more fuzzy. Still, it seems to be a reasonable baseline to demand for the $Core_c(o)$ to contain a great (here: at least a half) part of the objects over confirmed by the society as c.

Various meta-modifications can be analyzed and applied to the process of the $Core$ evaluation:

- in order to avoid meaningless (almost empty) cores, a core generated by the prototype candidate needs to contain at least a certain number of positive examples,
- in order to strengthen the influence of objects appearing more often in the environment, an overall cardinality of confirming/rejecting learning episodes could be reflected when evaluating core's support (e.g. imagine a case where a set of 3 commonly appearing objects is consistently confirmed as c while there are 10 other rarely appearing objects getting inconsistent labeling – it could be, in particular, related to a typical better alignment of the language in an area of frequent or better known items),
- in order to potentially reduce a complexity of the solution, the first prototype candidate generating a 'good' enough core is accepted (clearly visible in the `while` condition of the algorithm 4.1); it is in particular interesting how far is an accepted candidate from the best candidate in relation to a chosen core evaluation strategy,
- in order to deal with inconsistencies in a learning population, a certain number of negative examples might be allowed into the core as long as they are overwhelmed by a large number of positive examples; in such a case a different method of radii evaluation would be needed,
- in order to better reflect a distribution of the learning experience, values of radii τ_c^+, τ_c^- might be readjusted.

More excessive modifications to the overall strategy could involve:

- a density-based analysis of a category usage and a generation of a set of sub-prototypes for respective dense sub-clusters of objects in the conceptual space; in the case of more complex categories an application of single-sphere approach might not be enough; still, although it is feasible in general, in the dissertation such an approach is avoided in order to make the process computationally feasible in runtime,

- a more complex underlying structure of categories in the sense that instead of relying just on a function for a comparison of objects, the categories could be derived from an internal structure of the object themselves; it surely is an interesting direction for further studies.

The above modifications should keep the condition (4.1.1). The proof below shows that it is true for the algorithm 4.2.

Theorem 4.1. *If $\tau_c^+ \neq NULL$ and $\tau_c^- \neq NULL$, the thresholds τ_c^+ and τ_c^- calculated by the algorithm 4.2 satisfy the condition*

$$\tau_c^+ < \tau_c^-.$$

Proof. According to line 14 of the algorithm

$$\tau_c^+ = \max\{f \in F^+\} \quad \text{if} \quad F^+ \neq \varnothing$$

and according to line 13

$$F^+ := \{f(o^+, o) : o^+ \in E^+ \wedge f(o^+, o) < f^-_{\min}(o)\}.$$

From the above it is known that the value of τ_c^+ is equal to the distance of some object $o^+ \in E^+$ from the selected object $o \in O$. The above value must be smaller than the distance of any object $o^- \in E^-$ from the selected object $o \in O$, because according to line 11

$$f^-_{\min}(o) := \min_{o^- \in E^-} \{f(o^-, o)\}.$$

In turn, the value of τ_c^- is equal to the distance of some object $o^- \in E^-$ from the selected object $o \in O$, because according to line 16

$$\tau_c^- = \min\{f \in F^-\} \quad \text{if} \quad F^- \neq \varnothing$$

and according to line 15

$$F^- := \{f(o^-, o) : o^- \in E^- \wedge f(o^-, o) > f^+_{\max}(o)\}.$$

It follows from the above that

$$\tau_c^+ < \tau_c^-.$$

■

4.4.2 Computational example

Let the cognitive universe O consist of 16 objects, namely $O = \{o_1, ..., o_{16}\}$. The following learning multiset be given[1]: $Exp_c = \{1(o_1,+)^c, 1(o_3,+)^c, 2(o_4,+)^c, 2(o_6,+)^c, 1(o_6,-)^c, 2(o_{10},-)^c, 1(o_{13},-)^c, 3(o_{14},-)^c\}$ and the macrostructure f is a distance function.

Execution of the strategy for the given input and the assumed macrostructure leads to the following results.

Initial computations

- $E^+ = E_c^+(Exp_c) = \{o_1, o_3, o_4, o_6\}$,
- $E^- = E_c^-(Exp_c) = \{o_6, o_{10}, o_{13}, o_{14}\}$,
- $\hat{E}_c^+(Exp_c) = \{1o_1, 0o_2, 1o_3, 2o_4, 0o_5, 2o_6, 0o_7, 0o_8, 0o_9, 0o_{10}, 0o_{11}, 0o_{12}, 0o_{13}, 0o_{14}, 0o_{15}, 0o_{16}\}$.

The first step is to calculate the set of candidates for prototype, basing on $\hat{E}_c^+(Exp_c)$ and macrostructure. Let us assume, that procedure *extractCandidates* returns the set $Candidates = \{o_2, o_4\}$. Let us note, that object o_2 did not appear in agent's experience (multiplicity of o_2 in $\hat{E}_c^+(Exp_c)$ is equal to zero), so it is imaginary object from agent's point of view. Such an object would be rejected from *Candidates* if we would impose that all candidates have to be medoids.

Iteration 1 Let $o_2 \in Candidates$ be chosen as a candidate prototype o, i.e. further $o = o_2$ and $Candidates = \{o_4\}$. Next, distance values $f(o_2, o^+)$ for all $o^+ \in E^+$ and $f(o_2, o^-)$ for all $o^- \in E^-$ are computed. Let us assume,that their values are as follows:

- $f(o_2, o_1) = 1, f(o_2, o_3) = 2, f(o_2, o_4) = 1, f(o_2, o_6) = 1$,
- $f(o_2, o_6) = 1, f(o_2, o_{10}) = 1, f(o_2, o_{13}) = 3, f(o_2, o_{14}) = 2$.

On this basis, we determine in turn $f^-_{\min}(o_2) = 1$, $f^+_{\max}(o_2) = 2$, $F^+ = \varnothing$, $\tau_c^+ = NULL$, $F^- = \{3\}$, and $\tau_c^- = 3$, which leads to:

- $Core_c(o_2) = \varnothing$,
- $Outer_c(o_2) = \{o_{13}\}$,
- $Boundary_c(o_2) = \{o_1, o_3, o_4, o_6, o_{10}, o_{14}\}$.

The results achieved do not meet the condition for category c to be learned because $|Core_c(o_2)| = 0 < 4 = |Boundary_c(o_2) \cap E^+|$. Since the set *Candidates* is not empty, the next iteration is possible.

[1]elements with multiplicity $k_o = 0$ are ommited

Iteration 2 Let the only object $o_4 \in Candidates$ be chosen as a candidate prototype o, i.e. further $o = o_4$ and $Candidates = \varnothing$. Again, distance values $f(o_4, o^+)$ for all $o^+ \in E^+$ and $f(o_4, o^-)$ for all $o^- \in E^-$ are computed. Let us assume,that their values are as follows:

- $f(o_4, o_1) = 2, f(o_4, o_3) = 1, f(o_4, o_4) = 0, f(o_4, o_6) = 2,$
- $f(o_4, o_6) = 2, f(o_4, o_{10}) = 2, f(o_4, o_{13}) = 4, f(o_4, o_{14}) = 3.$

On this basis, we determine in turn $f^-_{\min}(o_4) = 2$, $f^+_{\max}(o_4) = 2$, $F^+ = \{0, 1\}$, $\tau_c^+ = 1$, $F^- = \{3, 4\}$, and $\tau_c^- = 3$, which leads to:

- $Core_c(o_4) = \{o_3, o_4\}$,
- $Outer_c(o_4) = \{o_{13}, o_{14}\}$,
- $Boundary_c(o_4) = \{o_1, o_6, o_{10}\}$.

In this case, the results achieved meet the condition for category c to be learned because $|Core_c(o_2)| = 2 \geqslant 2 = |Boundary_c(o_2) \cap E^+|$. In consequence, the object o_4 is assigned as the prototype $o_c^\star$ of the properly established category c with τ_c^+ and τ_c^- as its radii. The model of category c is well-defined and can be integrated with the ontological knowledge base.

4.4.3 Scheme for evaluation of computational complexity of strategy

Theorem 4.2. *The computational complexity of the algorithm 4.2 is of the order:*

$$O(\sum_t |X_t| + |Exp_c| + e + C \cdot p \cdot (|E^+| + |E^-|))$$

where

- X_t *– a set of objects in* $Episode(t)$*,*
- e *– computational complexity of the* $extractCandidates$ *function,*
- $C = |Candidates|$*,*
- p *– computational complexity of the expression* $f(o_i, o_j)$*.*

Proof. The first step of the algorithm is preprocessing the data collected by the agent in subsequent episodes to the Exp_c multiset (line 1). The computational complexity of this step is proportional to the number of objects observed in all the episodes considered. If we denote the set of objects in the $Episode(t)$ by X_t, the computational complexity will be of the order $O(\sum_t |X_t|)$. The next steps of the algorithm (lines 3 and 4) require inspection of the Exp_c multiset in order to find the sets E^+ and E^-. The size of the Exp_c multiset is $|Exp_c| = 2|O|$. In turn, the size of the set O grows exponentially with the size of the set of attributes, because $|O| = \prod_{a \in A} |V_a|$. In a pessimistic case, finding the sets E^+ and E^- can therefore have computational and memory complexity exponentially dependent on $|A|$. In practice, however, we expect that a very small fraction of the elements belonging to Exp_c have the multiplicity $k_o > 0$. Hence, it is convenient to store in the memory only elements of Exp_c for which the multiplicity $k_o > 0$. This should significantly reduce the memory requirements and the number of operations needed to review the Exp_c multiset.

On line 5, the *extractCandidates* function is executed. Its complexity can be very different depending on the adopted macrostructure and field of application. In the chapter 6 we will analyze the complexity of the *extractCandidates* function for the various implementation models. Here it is simply denoted by e and treated as a parameter of the formula for the computational complexity of the whole algorithm.

The number of iterations of the while loop (line 6) depends on the number of candidates. We will abbreviate it with $C = |Candidates|$. In the worst case, it will be equal to the size of the O set, but in practice the *extractCandidates* function should return a much smaller set of candidates.

Inside the loop, in line 9 the macrostructure value $f(o, o^+)$ needs to be calculated for all $o^+ \in E^+$. Again, the calculation of $f(o, o^+)$ strongly depends on the adopted macrostructure and the field of application. As for the *extractCandidates* function, in the chapter 6 we will analyze the computational complexity of the macrostructure calculation for various implementation models. Denoting by p the computational complexity of the expression $f(o_i, o_j)$, to complete the instruction from line 9, $p \cdot |E^+|$ operations need to be executed. Similarly, it takes $p \cdot |E^-|$ operations to execute instruction from line 10.

The complexity of the other instructions inside the loop (lines 11 to 28) is linearly dependent on the size of the sets E^+ and E^-. In summary, the computational complexity of the while loop is of the order $O(C \cdot p \cdot (|E^+| + |E^-|))$.

The final complexity of the entire algorithm results from the summation of the above estimates. ■

The theorem 4.2 shows that the computational complexity of the algorithm 4.2 is polynomial with respect to the number of objects observed by the agent in all episodes $\sum_t |X_t|$ and the number of attributes describing them $|A|$, when the following conditions are met:

- the size of the *Candidates* set does not grow exponentially with the number of attributes,

- the computational complexity of the expression $f(o_i, o_j)$ does not increase exponentially with the number of attributes,
- the computational complexity of the *extractCandidates* function does not increase exponentially with the number of objects or attributes,
- the size of the practically used part of the Exp_c multiset (i.e. elements with the multiplicity $k_o > 0$) does not increase exponentially with the number of attributes.

The above conditions determine the area of practical application of this model.

4.5 Connection to psycholinguistic theories

The six basic theses (T1-T6) of the standard version of the prototype semantics were introduced in chapter 3.2. The strategy and cognitive model described in this chapter are concordant with above theses, which is shown in detail below.

ad T1. The category has an internal prototype structure.

ad T2. The degree of representativeness of a given item needs to correspond to the degree of its membership to a category. In this model, belonging to a category is determined, among other things, on the basis of the distance from the prototype. The more representative elements are those closer to the prototype, and they are more likely to be included in the category, including its core.

ad T3. The elements of a given category do not have to possess properties common to all elements. In this model, category elements are connected to the prototype. The model can be extended in the future to include connections between elements.

ad T4. The boundaries of categories or concepts ought to be fuzzy. In this model, the category boundary contains elements that may or may not belong to a category.

ad T5. The belonging to a given category needs to be based on the degree of similarity to the prototype. In this work, the measures of distance from the prototype are considered, due to their easier implementation. However, distance can simply be thought of as the inverse measure of similarity, and so they can be used interchangeably.

ad T6. The belonging to a category should not be determined in an analytical manner, but in a holistic manner. In this model one does not analyze sets of necessary and sufficient conditions for the attributes of objects, like in classical definitions of categories. Instead, a more holistic measure of distance to the prototype is computed.

The numerical example described below corresponds to the example of category structure from psycholinguistic research, already shown in the figure 3.2. Thus, it can be seen that the model presented in this work enables the study of structures postulated in psycholinguistic research.

The table 4.2 shows an example environment experienced by the agent. For simplicity, data for one episode are shown. The data in the table for $x_{1,1}$, ..., $x_{1,5}$ objects correspond to the dependencies shown in the figure 3.2. The whole table can therefore be interpreted in such a way that the attributes a_1, ..., a_7 correspond to the features of different species of birds (1 means feature presence, 0 means no feature), and objects from $x_{1,1}$ to $x_{1,5}$ represent one individual from different species of birds.

episode	**object**	a_1	a_2	a_3	a_4	a_5	a_6	a_7	**labels**
	$x_{1,1}$	1	1	1	1	1	1	1	*is-bird*
	$x_{1,2}$	0	0	1	0	1	1	1	*is-bird*
	$x_{1,3}$	0	1	1	1	1	1	1	*is-bird*
$Episode(t_1)$	$x_{1,4}$	0	0	0	1	1	1	1	*is-bird*
	$x_{1,5}$	0	1	1	1	0	1	1	*is-bird*
	$x_{1,6}$	0	0	1	1	0	1	1	*not-bird*
	$x_{1,7}$	0	0	0	0	1	1	1	*not-bird*
	$x_{1,8}$	0	0	0	0	0	1	1	*not-bird*

Table 4.2: Episode for psycholinguistic example.

For the above data, the execution of the algorithm 4.2 will be shown.

The universe O is a set of binary vectors of length $n = 7$, thus $O = \{o_1, ..., o_{128}\}$. The multiset Exp_{bird} aggregating knowledge from all three episodes is as follows[2]:

$Exp_{bird} = \{1(o_{16}, +)^{bird}, 1(o_{24}, +)^{bird}, 1(o_{60}, +)^{bird}, 1(o_{64}, +)^{bird}, 1(o_{128}, +)^{bird}, 1(o_4, -)^{bird}, 1(o_8, -)^{bird}, 1(o_{28}, -)^{bird}\}$

where $o_4 = [0000011], o_8 = [0000111], o_{16} = [0001111], o_{24} = [0010111], o_{28} = [0011011], o_{60} = [0111011], o_{64} = [0111111], o_{128} = [1111111]$.

For the above data:

- $E^+ = E^+_{bird}(Exp_{bird}) = \{o_{16}, o_{24}, o_{60}, o_{64}, o_{128}\}$,
- $E^- = E^-_{bird}(Exp_{bird}) = \{o_4, o_8, o_{28}\}$.

Suppose the *extractCandidates* procedure has determined a one-element set of candidates $Candidates = \{o_{128}\}$, and the object distances from the candidate are as follows:

[2] the elements for which $k_{o_i} = 0$ are omitted

$f(o_4, o_{128}) = 5, \quad f(o_8, o_{128}) = 4, \quad f(o_{16}, o_{128}) = 3, \quad f(o_{24}, o_{128}) = 3,$
$f(o_{28}, o_{128}) = 3, \quad f(o_{60}, o_{128}) = 2, \quad f(o_{64}, o_{128}) = 1, \quad f(o_{128}, o_{128}) = 0.$

The above data can be presented graphically as shown in the figure 4.4. The situation shown in it is an extension of the case from the figure 4.3. Objects that belong to the set E^+, are marked in the figure with the symbol $\oplus$, and those belonging to the set E^- with the symbol $\ominus$. In square brackets, the distances of the objects from the candidate o_{128} are given.

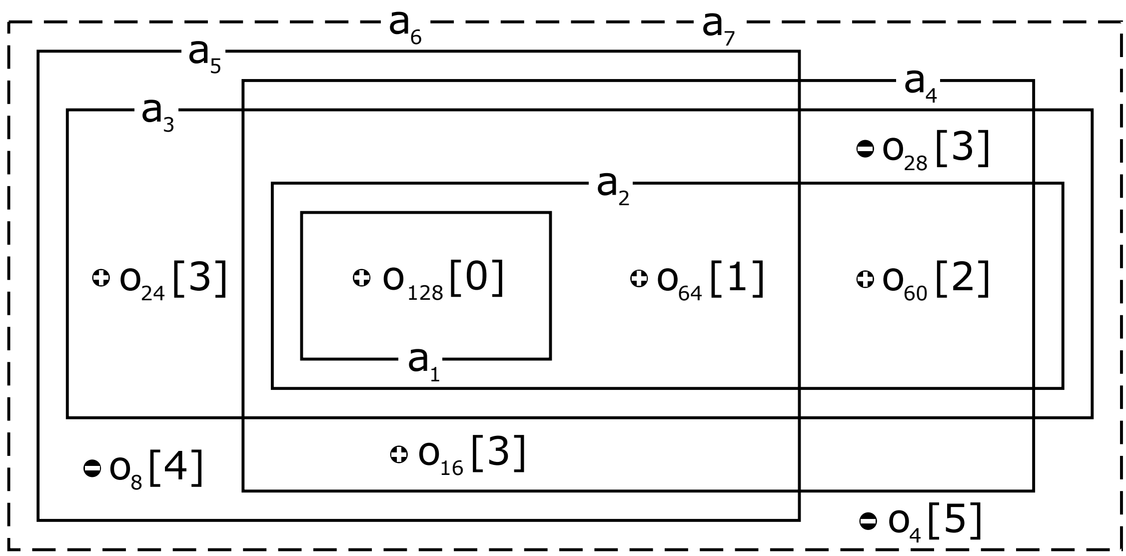


Figure 4.4: Exemplary cognitive model m_{bird}.

On this basis, we determine in turn $f^-_{\min}(o_{128}) = 3$, $f^+_{\max}(o_{128}) = 3$, $F^+ = \{2, 1, 0\}$, $\tau^+_{bird} = 2$, $F^- = \{5, 4\}$, and $\tau^-_{bird} = 4$, which leads to:

- $Core_{bird}(o_{128}) = \{o_{60}, o_{64}, o_{128}\}$,
- $Outer_{bird}(o_{128}) = \{o_4, o_8\}$,
- $Boundary_{bird}(o_{128}) = \{o_{16}, o_{24}, o_{28}\}$.

In this case, the results achieved meet the condition for category *bird* to be learned because $|Core_{bird}(o_{128})| = 3 \geqslant 2 = |Boundary_{bird}(o_{128}) \cap E^+|$. In consequence, the object o_{128} is assigned as the prototype $o^\star_{bird}$ of the properly established category *bird* with τ^+_{bird} and τ^-_{bird} as its radii. The model of category *bird* is well-defined and can be integrated with the ontological knowledge base.

The above sets and threshold values can be presented graphically as shown in the figure 4.5. The object's distance from the prototype in the drawing plane corresponds to the distance f from the prototype. The different directions of objects in relation to the prototype symbolize different values of the object's attributes. It should be emphasized, however, that the figure does not show the two-dimensional space of numerical attributes, but it is an attempt to show the multidimensional space of binary attributes, taking into account an additional dimension – the distance from the prototype. The figure cannot therefore be interpreted directly.

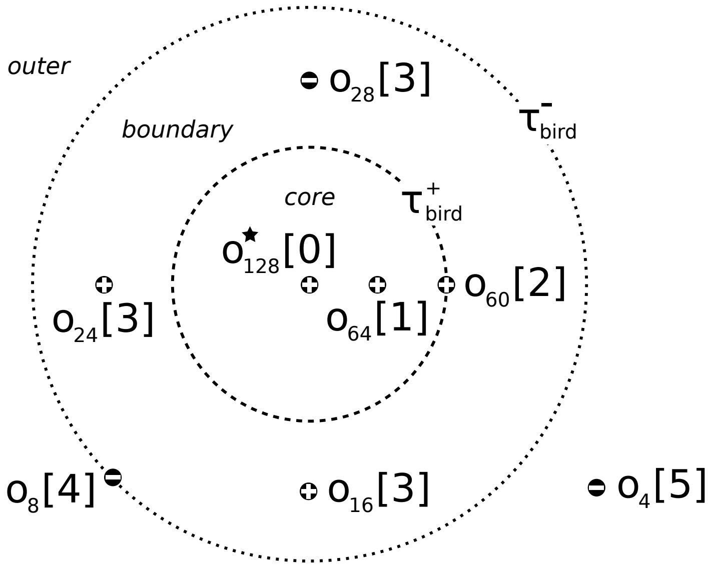


Figure 4.5: Model m_{bird} divided into three regions.

As in psycholinguistic experiments, we can find the best representative of the *bird* category – it is the object most similar (closest) to the prototype. In this case, it is o_{128}, which just happens to be the prototype. We can also rank the elements belonging to the category model according to the degree of their representativeness/belonging, they are sequentially: o_{128}, o_{64}, o_{60}, $\{o_{16}, o_{24}, o_{28}\}$, o_8 and o_4.

It is true that the information about the association of objects o_i from the mental space with objects x_{ij} from the working memory of the agent is lost during preprocessing, but we can still arrange them similarly. Thus, the ranking of the objects according to their degree of representativeness/belonging will be as follows: $x_{1,1}$, $x_{1,3}$, $x_{1,5}$, $\{x_{1,2}, x_{1,4}, x_{1,6}\}$, $x_{1,7}$ and $x_{1,8}$.

4.6 Ill-defined model of category

If none of the candidates meet the acceptance condition, the agent is unable to create a properly defined category model (learn the category). Such a model of category is called the ill-defined model. This situation can be obtained in the example from the previous chapter, if, for example, we remove the observations of objects $x_{1,3}$ and $x_{1,5}$ from the agent's experience. Then the multiplicities in the multi-set Exp_{bird} will change[3]:

$Exp_{bird} = \{1(o_{16},+)^{bird}, 1(o_{24},+)^{bird}, 1(o_{128},+)^{bird}, 1(o_4,-)^{bird}, 1(o_8,-)^{bird}, 1(o_{28},-)^{bird}\}$

For the above data:

- $E^+ = E^+_{bird}(Exp_{bird}) = \{o_{16}, o_{24}, o_{128}\}$,
- $E^- = E^-_{bird}(Exp_{bird}) = \{o_4, o_8, o_{28}\}$.

Assuming again, that the only candidate for a prototype is o_{128}, we determine $f^-_{\min}(o_{128}) = 3$, $f^+_{\max}(o_{128}) = 3$, $F^+ = \{0\}$, $\tau^+_{bird} = 0$, $F^- = \{5, 4\}$, and $\tau^-_{bird} = 4$, which leads to:

- $Core_{bird}(o_{128}) = \{o_{128}\}$,
- $Outer_{bird}(o_{128}) = \{o_4, o_8\}$,
- $Boundary_{bird}(o_{128}) = \{o_{16}, o_{24}, o_{28}\}$.

The acceptance condition will not be met because

$$|Core_{bird}(o_{128})| = 1 < 2 = |Boundary_{bird}(o_{128}) \cap E^+|.$$

This means that the quality of the knowledge obtained about the category c is too low for the agent to use its model. The figure 4.6 graphically illustrates this situation.

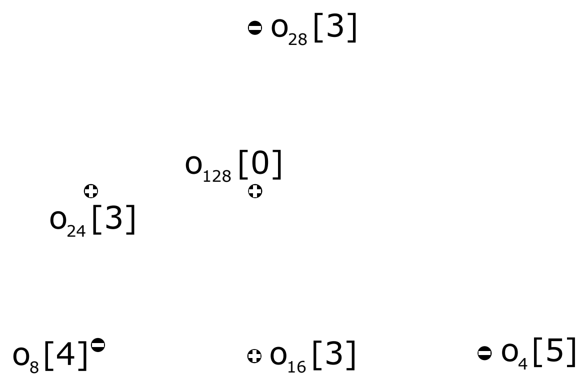


Figure 4.6: Ill-defined model of category.

[3]elements for which $k_{o_i} = 0$ are omitted

Chapter 5

Cognitive semantics of atomic statements

5.1 Language of modal categorization

In this chapter, we will consider the part of cognitive semantics, that deals with statements concerning the membership of currently observed (in the current episode) objects to categories known to the agent.

The basic statements about the membership of the object x to the category c are called **atomic class-membership statements**:

- "Object x belongs to category c."
- "Object x does not belong to category c."

The form of atomic statements suggests that they represent objective knowledge independent of the subject expressing them. This is an approach often taken in classical semantics – assuming the existence of certain idealized categories with which the objects of the real world can be matched in an unambiguous and objective manner. In this dissertation, the approach is adopted in which generated statements must relate to the knowledge of the subject expressing them. Therefore, we will take into account epistemic modalities, described in more detail in chapter 3.3. For that reason, we extend atomic statements with (epistemic) modal operators, thus obtaining **modal class-membership statements** expressing the belief of the subject (agent) about the truthfulness of the information transmitted:

- knowledge: "I know that " + [atomic statement],
- belief: "I believe that " + [atomic statement],
- possibility: "I find it possible that " + [atomic statement].

The use of the knowledge modal operator means that the agent is sure about the belonging (or not) of the object x to the category c. This certainty comes from the agent's experience, and so all knowledge accumulated by the agent must support this belief. In other words, the agent could not experience a situation that would contradict this belief, and is also unable to "imagine" such a situation. By imagining a situation, we mean the generation, within the agent's cognitive processes, of a set of attribute values that never appeared in the reality experienced by the agent. We could write the statements with the modal knowledge operator in a more elaborate form, emphasizing their epistemicity:

- "According to all **my** collected experience, **I am** certain that object x belongs to category c."
- "According to all **my** collected experience, **I am** certain that object x does not belong to category c."

The use of the belief modal operator means that the agent is not sure about the belonging (or not) of the object x to the category c, but it considers the appropriate state of affairs to be most likely. Thus, the agent may have experienced (probably few) situations that would contradict this belief, or at least is able to "imagine" such a situation. Statements with the modal belief operator in a more extensive form would look like this:

- "According to all **my** collected experience, **I** believe that object x belongs to category c."
- "According to all **my** collected experience, **I** believe that object x does not belong to category c."

The use of the possibility modal operator means that the agent is not sure about the belonging (or not) of the object x to the category c and that the degree of this uncertainty is quite high. So, the agent probably has experienced situations that would contradict this belief or is able to easily "imagine" such a situation. Statements with the modal possibility operator in a more extensive form would look like this:

- "According to all **my** collected experience, **I** find it possible that object x belongs to category c."
- "According to all **my** collected experience, **I** find it possible that object x does not belong to category c."

Keeping in mind the extended meaning of the above statements, they are presented below in a more formalized way, using the K modal categorization language.

Definition 5.1. The alphabet of the modal categorization language consists of the following elements:

- symbols $\{x_{t,1}, x_{t,2}, ...\} \in X_t$ representing objects in the agent's working memory, unequivocally pointing to the entities in the environment of the agent,
- set of category names C; the names are string literals e.g. *bird*, *robin*,
- symbol $\in$ for the binary relation specified on the set $X_t \times C$,
- symbol $\notin$ for the binary relation specified on the set $X_t \times C$,
- symbols Pos, Bel, $Know$ for unary modal operators of possibility, belief and knowledge,
- auxiliary symbols '(' and ')'.

To define the modal categorization language K, let's first define a set of non-modal atomic formulas K^N.

Definition 5.2. The syntactic structure of non-modal atomic formulas belonging to the K^N language is given as follows:

- each string of the form "$x \in c$" where $x \in X_t$ and $c \in C$, is a valid non-modal atomic language K^N formula,
- each string of the form "$x \notin c$" where $x \in X_t$ and $c \in C$, is a valid non-modal atomic language K^N formula.

After applying modal operators of possibility, belief and knowledge to non-modal atomic formulas, we obtain modal atomic formulas that make up the K language.

Definition 5.3. The semantic modal categorization language K is given as follows:

$$K = \{Know(\varphi) : \varphi \in K^N\} \cup \{Bel(\varphi) : \varphi \in K^N\} \cup \{Pos(\varphi) : \varphi \in K^N\}.$$

The intuitive semantics of non-modal and modal atomic formulas is presented in the tables below.

formula	**intuitive meaning**
$x \in c$	Object x belongs to category c.
$x \notin c$	Object x does not belong to category c.

Table 5.1: Intuitive semantics of non-modal atomic formulas.

formula	intuitive meaning
$Know(x \in c)$	I know that object x belongs to category c.
$Know(x \notin c)$	I know that object x does not belong to category c.
$Bel(x \in c)$	I believe that object x belongs to category c.
$Bel(x \notin c)$	I believe that object x does not belong to category c.
$Pos(x \in c)$	I find it possible that object x belongs to category c.
$Pos(x \notin c)$	I find it possible that object x does not belong to category c.

Table 5.2: Intuitive semantics of modal atomic formulas.

5.2 Complete representation of object

Let us recall that the agent's actions in each episode can be summarized as follows:

1. The agent determines what objects are in its environment and what are the values of the attributes that describe them.
2. The agent determines what labels given by the teacher are attached to the objects.
3. For each object-label association, the agent updates the cognitive structure that describes the category that the label points to.
4. The agent generates grounded statements about the categorization of objects for selected objects in the agent's environment.

In this subchapter the fourth point is described in detail, i.e. the grounding of the statement by the agent, in case when the values of all the attributes of the considered object are known, i.e. for the selected object x is met $(\forall a \in A)(percept(x, a) \in V_a)$. Statements will be generated for the cognitive state of $SP(t)$ at a point in time t.

Definition 5.4. At each time point $t \in T$ the t-related state of cognitive processes of agent is described by the following set

$$SP(t) = \{M, Episode(t)\}$$

where M is set of models of categories in agent's ontology and $Episode(t)$ is episode in time point t available to agent's perception.

The rules for grounding of statements will be presented in two equivalent ways:

- in procedural form as algorithm 5.1,
- as conditions of epistemic satisfaction relation, introduced in chapter 3.3.

Both algorithm and epistemic satisfaction relation definitions assume, that macrostructure is a distance function. If macrostructure is a similarity function, then the formalizations need to be redefined accordingly.

At the input of the statements generation algorithm, there is the t-related cognitive state consisting of a set of models belonging to the agent's cognitive structures and an episode including observations of objects in the agent's environment. The result of the algorithm is a set of statements S_t generated by the agent about the membership of objects to the category. The statements refer to the current state of the environment (in time point t), therefore the set of statements is marked with the index t. Statements from the above set should be conveyed to the participants of the interaction using the means of communication available to the agent, e.g. in text or audio form. The algorithm generates a set of all valid statements concerning all objects available to agent's perception, but in practical situations probably only subset of S_t would be provided to other participants of communication. This subset would contain the statements relevant to present context of communication, for example guided by questions like "*What is this object?*" or "*Does this object belong to category c?*".

The main loop processes all objects observed in the episode. The function $Preprocess$ finds object $o \in O$ corresponding to object x (line 2). This operation is required, because the grounding of statements is based on location of objects in mental space, and not on their observed realizations. It was already described in more detail in chapter 4.2.2.

The object o is matched to all well-defined category models found in the agent's cognitive structures. The omission of ill-defined category models is necessary to increase the agent's credibility in the subjects to which his statements are addressed (line 4). For ill-defined categories, the agent will not generate statements about the object's membership to the category.

For each of the models, one or two epistemic modal atomic formulas are generated and added to the set of statements S_t (lines 5-17). The form of the above formulas depends on which of the model areas of a given category the object o will be classified into. The object is assigned to a given area on the basis of its distance to the category prototype. Example of grounding of statements for two categories is shown in the figure 5.1. The object $x_{2,1}$ is observed realization of mental object o_7. Object o_7 is located in the core of model of category *bird*, which is a premise for grounding the statement expressing high certainty about the membership of the object $x_{2,1}$ to the category *bird*. It is also located in the boundary of model of category *eagle*, which is a premise for grounding the statement expressing low certainty about the membership of the object $x_{2,1}$ to the category *eagle*.

In general, we consider the core of the category's model m_c to include objects that most certainly belong to the category c. Therefore, including an object in the core

Algorithm 5.1: Grounding of statements.

Input: state of cognitive processes $SP(t) = \{M,\, Episode(t)\}$.
Output: set of statements S_t.

foreach *object* x *in* X_t **do**
  $o :=$ Preprocess(x);
  `// generate statements`
  foreach *model* m_c *in* M **do**
    if m_c *is well* $-$ *defined* **then**
      if $f(o, o_c^\star) \leqslant \tau_c^+$ **then**
        `// object fits core`
        $S_t := S_t \cup \{Know(x \in c)\}$;
      else if $f(o, o_c^\star) \geqslant \tau_c^-$ **then**
        `// object fits outer`
        $S_t := S_t \cup \{Know(x \notin c)\}$;
      else
        `// object fits boundary`
        $\varepsilon := ER(\tau_c^-, \tau_c^+)$;
        $EN_c(o, \varepsilon) := \{e \in E_c^+ \cup E_c^- : f(e, o) \leqslant \varepsilon\}$;
        if $\lambda_c(EN_c(o, \varepsilon)) \geqslant \lambda_{minBel}$ **then**
          $S_t := S_t \cup \{Bel(x \in c)\}$;
          $S_t := S_t \cup \{Pos(x \notin c)\}$;
        else
          $S_t := S_t \cup \{Bel(x \notin c)\}$;
          $S_t := S_t \cup \{Pos(x \in c)\}$;

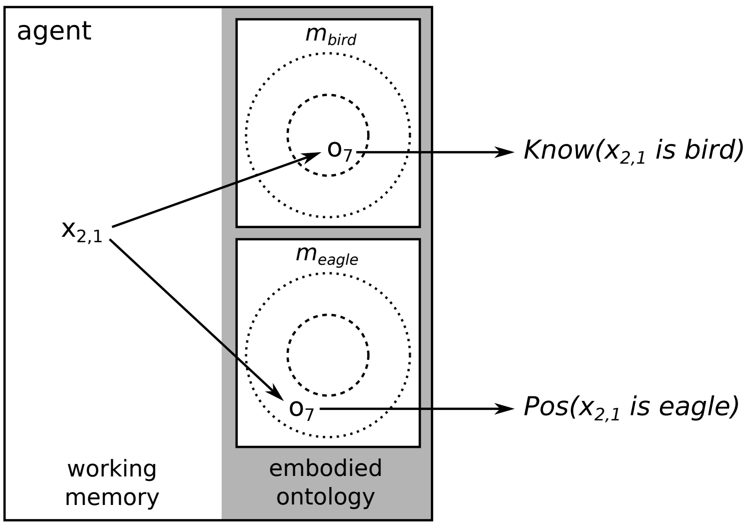


Figure 5.1: Grounding of statements.

of the category (lines 5 to 6) is the basis for grounding the statement, the intuitive meaning of which can be expressed as "I know that object x belongs to category c."

Definition 5.5. Let the time point t and the state of cognitive processes $SP(t)$ described by the episode *Episode*(t) and the set of cognitive models M containing the well-defined model m_c be given. For each object $x \in X_t$ and category c we assume that the epistemic satisfaction relation $SP(t) \vDash_G Know(x \in c)$ holds if and only if

$$f(o, o_c^\star) \leqslant \tau_c^+$$

where object x is observed realization of mental object o in time point t.

We recognize that the outer region of category's model m_c includes objects that are definitely not included in the category c. Therefore, including an object in the outer region of the category (lines 7 to 8) is the basis for grounding the statement, the intuitive meaning of which can be expressed as "I know that object x does not belong to category c."

Definition 5.6. Let the time point t and the state of cognitive processes $SP(t)$ described by the episode *Episode*(t) and the set of cognitive models M containing the well-defined model m_c be given. For each object $x \in X_t$ and category c we assume that the epistemic satisfaction relation $SP(t) \vDash_G Know(x \notin c)$ holds if and only if

$$f(o, o_c^\star) \geqslant \tau_c^-$$

where object x is observed realization of mental object o in time point t.

We consider that the boundary of category's model m_c includes objects that may or may not belong to the category c. Therefore, including an object in the boundary of category (lines 10 to 17) is the basis for establishing modal statements with operators of beliefs and possibilities, the intuitive meaning of which can be expressed as "I believe that object x belongs/does not belong to category c." or "I find it possible that object x belongs/does not belong to category c."

The statements with the knowledge operator are grounded only on the basis of the distance from the category prototype, which is equivalent to examining belonging to the appropriate region of the category's model – the core or the outer region of the category. In case of operators of belief and possibility, the distance to the prototype is not a sufficient basis for determining the degree of certainty of the verbal judgment regarding membership to a category. To define it, we will use the concept of the **relative grounding strength**, determined by the distance of the considered object from both positive and negative pieces of information in the model.

Definition 5.7. For object $o \in O$ by epistemic neighborhood EN_c we understand a set of objects defined as follows:

$$EN_c(o, \varepsilon) = \{e \in (E_c^+(Exp_c) \cup E_c^-(Exp_c)) : f(e, o) \leqslant \varepsilon\}$$

where $\varepsilon \in \mathbb{R}$ is called the **radius of the epistemic neighborhood**.

The radius of the epistemic neighborhood ε can be determined in various ways, for example it can be an experimentally chosen constant. In this dissertation, it is considered that the radius is determined by the function ER, depending on the value of thresholds τ_c^- and τ_c^+ delineating the category regions. In the following considerations, we will assume that the value of the ER function depends linearly on the width of the boundary of the category model, i.e. $ER(\tau_c^-, \tau_c^+) = \alpha(\tau_c^- - \tau_c^+)$ where $\alpha \in \mathbb{R}$ is the coefficient of the radius of the epistemic neighborhood. Thanks to this, the greater the boundary of the category – and hence the greater the uncertainty as to whether an observation belongs to a category – the greater the epistemic neighborhood considered when grounding statements. A larger neighborhood will usually mean that the decision to select a modal operator will be made on the basis of more experience.

The next definition determines the concept of relative grounding strength.

Definition 5.8. For the set of objects $Q \subseteq (E_c^+(Exp_c) \cup E_c^-(Exp_c))$ relative grounding strength $\lambda_c(Q)$ is defined as follows:

$$\lambda_c(Q) = \begin{cases} 0 & if \quad |Q| = 0 \\ \frac{|Q \cap E_c^+(Exp_c)|}{|Q|} & if \quad |Q| > 0. \end{cases}$$

Thus, it is the ratio of the number of positive objects in Q to all objects in that set.

Definition 5.9. Let the time point t, the state of cognitive processes $SP(t)$ described by the episode $Episode(t)$ and the set of cognitive models M containing the well-defined model m_c, the radius of the epistemic neighborhood ε and $\lambda_{minBel} \in (0, 1]$ threshold be given. For any object $x \in X_t$ and category c we assume that epistemic satisfaction relations $SP(t) \vDash_G Bel(x \in c)$ and $SP(t) \vDash_G Pos(x \notin c)$ hold if and only if

$$\left(\tau_c^+ < f(o, o_c^\star) < \tau_c^-\right) \wedge \left(\lambda_c(EN_c(o, \varepsilon)) \geqslant \lambda_{minBel}\right)$$

where object x is observed realization of mental object o in time point t.

Definition 5.10. Let the time point t, the state of cognitive processes $SP(t)$ described by the episode $Episode(t)$ and the set of cognitive models M containing the well-defined model m_c, the radius of the epistemic neighborhood ε and $\lambda_{minBel} \in (0, 1]$ threshold be given. For any object $x \in X_t$ and category c we assume that epistemic satisfaction relations $SP(t) \vDash_G Bel(x \notin c)$ and $SP(t) \vDash_G Pos(x \in c)$ hold if and only if

$$\left(\tau_c^+ < f(o, o_c^\star) < \tau_c^-\right) \wedge \left(\lambda_c(EN_c(o, \varepsilon)) < \lambda_{minBel}\right)$$

where object x is observed realization of mental object o in time point t.

5.2.1 Illustrative examples

Let us suppose that for model of category c the thresholds are equal to $\tau_c^+ = 5$ and $\tau_c^- = 8$. In $Episode(t_3)$ two objects $x_{3,8}$ and $x_{3,9}$ appeared in the agent's range of perception. Suppose that in the agent's cognition process the object $x_{3,8}$ from working memory corresponds to the object o_8 in embodied ontology, and the object $x_{3,9}$ from working memory corresponds to the object o_9 in embodied ontology. The distances between the objects and the prototype are $f(o_8, o_c^\star) = 3$ and $f(o_9, o_c^\star) = 10$. The above situation is presented in the figure 5.2.

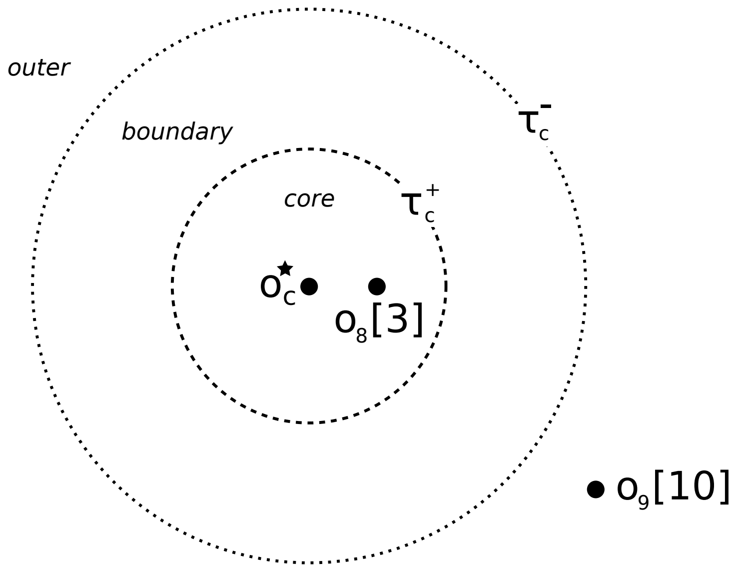


Figure 5.2: Grounding of objects in core and outer region of category model.

Since $f(o_8, o_c^\star) = 3 \leqslant \tau_c^+ = 5$ according to the definition 5.5 epistemic satisfaction relation holds for formula $Know(x_{3,8} \in c)$ and such a formula should be added to the set of statements S_t. The intuitive meaning of the formula can be expressed as "I know that object $x_{3,8}$ belongs to category c."

Since $f(o_9, o_c^\star) = 10 \geqslant \tau_c^- = 8$ according to the definition 5.6 epistemic satisfaction relation holds for formula $Know(x_{3,9} \notin c)$ and such a formula should be added to the set of statements S_t. The intuitive meaning of the formula can be expressed as "I know that object $x_{3,9}$ does not belong to category c."

A more complicated case is when an object is included in the boundary of the category. In order to establish the right statement, the agent must then compare the considered object not only with the prototype but also with other objects in embodied ontology. Let's make similar assumptions as in the previous example, except that this time the distances between the objects and the prototype are $f(o_8, o_c^\star) = 6$ and $f(o_9, o_c^\star) = 6$. The above situation is presented in the figure 5.3.

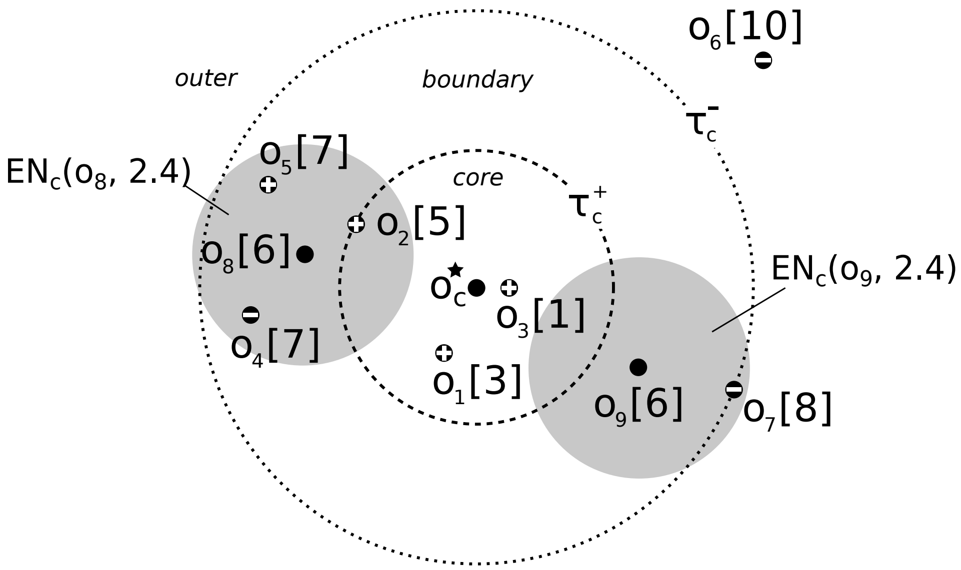


Figure 5.3: Grounding of objects in boundary of category model.

Since $\tau_c^+ = 5 < f(o_8, o_c^\star) = 6 < \tau_c^- = 8$ and $\tau_c^+ = 5 < f(o_9, o_c^\star) = 6 < \tau_c^- = 8$ the relative grounding strength must be determined for both objects. The first step is to establish the radius of the epistemic neighborhood ε. As already mentioned, we apply the formula for the linear dependence of the radius on the width of the boundary. Assuming $\alpha = 0.8$ we get $\varepsilon = ER(\tau_c^-, \tau_c^+) = \alpha(\tau_c^- - \tau_c^+) = 0.8 \cdot (8 - 5) = 2.4$

According to definition 5.7 we calculate the epistemic neighborhood of objects, i.e. $EN_c(o_8, \varepsilon)$ and $EN_c(o_9, \varepsilon)$. For this purpose, it is necessary to calculate the distance from the objects o_8 and o_9 to the objects in model of c in embodied ontology. Let us assume that the above distances are given as in the table 5.3 and that:

- $E_c^+(Exp_c) = \{o_1, o_2, o_3, o_5\}$
- $E_c^-(Exp_c) = \{o_4, o_6, o_7\}$.

o_i	o_1	o_2	o_3	o_4	o_5	o_6	o_7
$f(o_8, o_i)$	6	1	7	1	1	14	14
$f(o_9, o_i)$	6	11	5	11	13	8	2

Table 5.3: Distances between objects in the example for complete representation.

Based on the above data, we determine $EN_c(o_8, \varepsilon) = EN_c(o_8, 2.4) = \{o_2, o_4, o_5\}$ and $EN_c(o_9, \varepsilon) = EN_c(o_9, 2.4) = \{o_7\}$. According to the definition 5.8 we can calculate the relative grounding strength $\lambda_c(EN_c(o_8, 2.4)) = \frac{|\{o_2,o_5\}|}{|\{o_2,o_4,o_5\}|} = 2/3$, and also $\lambda_c(EN_c(o_9, 2.4)) = \frac{|\varnothing|}{|\{o_7\}|} = 0/1$.

Let us assume $\lambda_{minBel} = 0.5$. Such a threshold value means that if at least half of the elements in the epistemic neighborhood of the considered object are positive experiences, then the agent is willing to establish a statement with the operator of the belief that the object belongs to the category. On the other hand, if in the epistemic neighborhood of the considered object more than half of the elements are negative experiences, then the agent is willing to establish a statement with the operator of the possibility regarding the membership of the object to the category.

Since $\lambda_c(EN_c(o_8, 2.4)) = 2/3 \geqslant \lambda_{minBel} = 0.5$ according to the definition 5.9 epistemic satisfaction relation holds for formulas $Bel(x_{3,8} \in c)$ and $Pos(x_{3,8} \notin c)$ and such formulas should be added to the set of statements S_t. The intuitive meaning of the formulas can be expressed as "I believe that object $x_{3,8}$ belongs to category c." and "I find it possible that object $x_{3,8}$ does not belong to category c."

Since $\lambda_c(EN_c(o_9, 2.4)) = 0 < \lambda_{minBel} = 0.5$ according to the definition 5.10 epistemic satisfaction relation holds for formulas $Bel(x_{3,9} \notin c)$ and $Pos(x_{3,9} \in c)$ and such formulas should be added to the set of statements S_t. The intuitive meaning of the formulas can be expressed as "I believe that object $x_{3,9}$ does not belong to category c." and "I find it possible that object $x_{3,9}$ belongs to category c."

Note that the distance from o_8 and o_9 objects to the category's prototype is the same, but the statements generated by the agent are different due to the different neighborhood of each object.

5.3 Incomplete representation of object

As already mentioned, the perception of the agent is limited, that is, it is not always possible to determine the value of each attribute for each object. In this chapter, we consider the grounding of an agent's statement when some of the attribute values of the object under consideration may not be known. According to definition 4.1 the function *percept* : $X \times A \longrightarrow V$ is a total function, that is defined for all objects x and attributes a. We have to lift this limitation to model a situation where some of the attribute values for a given object are unknown. So, the definition of an extended episode would be as follows:

Definition 5.11. The extended episode, i.e. the state of the agent's environment at the moment t, is a tuple of the form $ExtEpisode(t) = \langle X_t, A, ExtV, L_t, extPercept, label\rangle$, where

- X_t – a finite set of objects,
- A – a finite set of attributes,
- $ExtV = \bigcup_{a\in A} V_a \cup \{\epsilon\}$, where V_a is the domain of attribute a, ϵ denotes the unknown value of the attribute,
- L_t – a finite set of labels,
- $extPercept$ – a function representing observed values of attributes of objects, $X_t \times A \longrightarrow ExtV$, such that $extPercept(x, a) \in V_a \cup \{\epsilon\}$ for all $x \in X_t$ and $a \in A$,
- $label$ – a function representing the observed assignments of labels to objects, $X_t \longrightarrow \Pi(L)$, $\Pi(L)$ is a power set of L.

Similarly, the mental space of objects in the agent's cognitive structures must be expanded. The $ExtO$ set is an extension of the O set with objects containing incomplete information about the values of their attributes:

Definition 5.12. Universe of mental representations of distinguishable objects with incomplete information $ExtO$ is defined as:

$$ExtO = \{o_1, ..., o_N\} \stackrel{\text{def}}{=} \bigtimes_{a\in A} \Big(V_a \cup \{\epsilon\}\Big)$$

The proposed strategy for dealing with a missing attribute value assumes that the missing value is replaced by one or more of the possible values for the attribute. It can be said that the agent tries to "imagine" what the observed object could look like if the object had some variants of a given attribute, and then – according to the rules of cognitive semantics – to which category the object would belong to. This means that a single object with incomplete information is replaced with one or more objects with complete information.

Formally, it can be expressed by a function which returns a set of objects with complete information for an object with incomplete information. We call this function $cReps$ for complete representation, hence $cReps : ExtO \longrightarrow \Pi(O)$. Algorithm 5.2 is an example implementation of the above function, in which the missing attribute values are replaced with all possible values of a given attribute.

Algorithm 5.2: Example of *cReps* function

Input: object $o \in ExtO$ with incomplete information.
Output: set of objects $R \in \Pi(O)$ with complete information.

$R := \{o\}$;
foreach *attribute a in A* **do**
  if $o(a) = \epsilon$ **then**
    $NewR := \varnothing$;
    foreach *value v in* V_a **do**
      foreach *object r in R* **do**
        $r' :=$ duplicate of r;
        $r'(a) := v$;
        add r' to $NewR$;
    $R := NewR$;

Remark 5.1. *If as an argument of the algorithm 5.2 we provide the object o with complete information, the result of the function will be a one-element set containing this object: cReps*$(o) = \{o\}$.

Example 3. Let us assume a microstructure of all objects o from a cognitive universe $ExtO$ in a form of fixed-length binary vectors, where length $n = 4$. Set of attributes is denoted as $A = (a_1, a_2, a_3, a_4)$. Let us consider object $o \in ExtO$ such that $o = (\epsilon, \epsilon, 0, 1)$. The result of algorithm 5.2 is a set *cReps*$(o) = \{(0,0,0,1), (0,1,0,1), (1,0,0,1), (1,1,0,1)\}$.

Obviously, the size of the set returned by algorithm 5.2 grows exponentially with a number of unknown attribute's values. For example if there are n binary attributes with unknown values in object o, then algorithm 5.2 returns a set of 2^n objects – with all possible combinations of attribute's values that the agent can "imagine".

Additional restrictions can be imposed on the *cReps* function, which in practice should limit the size of the set of objects returned by *cReps*. For example, for the function *cReps* to return only objects that:

- were actually observed by an agent, i.e. *cReps*$(o) \subseteq E_c^+(Exp_c) \cup E_c^-(Exp_c)$,
- meet the rule forbidding the simultaneous occurrence of certain values of attributes in the object description; for example, the agent may consider an animal having hair or feathers, but not both,
- meet the rule according to which the occurrence of a certain value of an attribute in the description of an object results in the occurrence of a certain value of another attribute; for example, the agent may "know" that if an animal has feathers, it must also have a beak.

The positioning of the observation of the object within the category model becomes more complicated as the object is then represented by a set of points, rather than a single point in mental space. Thus, cognitive semantics must be adapted to consider a set of objects. For each "imagined" version of the object, the value of the macrostructure can be calculated and used to place it in the right area of the model. More formally, the above ideas are presented in the algorithm 5.3.

Algorithm 5.3: Extended grounding of statements.

Input: state of cognitive processes $SPE(t) = \{M,\, ExtEpisode(t)\}$.
Output: set of statements S_t.

foreach *object* x *in* X_t **do**
  $o :=$ Preprocess(x);
  $R := cReps(o)$;
  `// generate statements`
  foreach *model* m_c *in* M **do**
    if m_c $is\ well-defined$ **then**
      if $(\forall r \in R)\ f(r, o_c^\star) \leqslant \tau_c^+$ **then**
        `// all objects fit core`
        $S_t := S_t \cup \{Know(x \in c)\}$;
      else if $(\forall r \in R)\ f(r, o_c^\star) \geqslant \tau_c^-$ **then**
        `// all objects fit outer`
        $S_t := S_t \cup \{Know(x \notin c)\}$;
      else
        `// some objects fit boundary`
        $\varepsilon := ER(\tau_c^-, \tau_c^+)$;
        $EN_c(R, \varepsilon) := \varnothing$;
        foreach *object* r *in* R **do**
          $EN_c(R, \varepsilon) := EN_c(R, \varepsilon) \cup \{e \in E_c^+ \cup E_c^- : f(e, r) \leqslant \varepsilon\}$;
        if $\lambda_c(EN_c(R, \varepsilon)) \geqslant \lambda_{minBel}$ **then**
          $S_t := S_t \cup \{Bel(x \in c)\}$;
          $S_t := S_t \cup \{Pos(x \notin c)\}$;
        else
          $S_t := S_t \cup \{Bel(x \notin c)\}$;
          $S_t := S_t \cup \{Pos(x \in c)\}$;

The statements are generated for the extended cognitive state $SPE(t)$ at a point in time t.

Definition 5.13. At each time point $t \in T$ the extended t-related state of cognitive processes of agent is described by the following set

$$SPE(t) = \{M, ExtEpisode(t)\}$$

where M is set of models of categories in agent's ontology and $ExtEpisode(t)$ is extended episode in time point t available to agent's perception.

The main loop processes all objects observed in the episode. The function $Preprocess$ finds object $o \in O$ corresponding to object x (line 2). This operation is required, because the grounding of statements is based on the location of objects in mental space, and not on their observed realizations. It was already described in more detail in chapter 4.2.2.

Then, for an object with incomplete information, a corresponding set of objects with complete information R is found (line 3). All objects $r \in R$ are matched to all well-defined category models found in the agent's cognitive structures. The omission of poorly defined category models (line 5) is necessary to increase the agent's credibility in the subjects to which his statements are addressed. For ill-defined categories, the agent will not generate statements regarding the membership of the object to the category.

For each of the models, one or two epistemic modal atomic formulas are generated and added to the set of statements S_t (lines from 6 to 20). The form of the above formulas depends on which of the model areas of a given category includes the objects in R. The object o is assigned to a given region based on the distance of the corresponding objects $r \in R$ to the category prototype.

Remark 5.2. *Algorithm 5.3 is the extended version of algorithm 5.1. It works both with objects with complete and incomplete information. In the first case, since* $cReps(o) = \{o\}$*, it will behave exactly as the algorithm 5.1.*

In general, we consider the core of the category's model m_c to include objects that most certainly belong to the category c. Therefore, including all of the objects $r \in R$ in the core of the category (lines 6 to 7) is the basis for grounding the statement, the intuitive meaning of which can be expressed as "I know that object x belongs to category c."

Definition 5.14. Let the time point t and the extended state of cognitive processes $SPE(t)$ described by the episode $ExtEpisode(t)$ and the set of cognitive models M containing the well-defined model m_c be given. For each object $x \in X_t$ and category c we assume that the epistemic satisfaction relation $SPE(t) \models_G Know(x \in c)$ holds if and only if

$$\forall r \in R \;\; f(r, o_c^{\star}) \leqslant \tau_c^{+}$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t. This definition is an extension of the definition 5.5 and can also be used for objects with complete information, since then $R = \{o\}$.

We recognize that the outer region of category's model m_c includes objects that are definitely not included in the category c. Therefore, including all of the objects $r \in R$ in the outer region of the category (lines 8 to 9) is the basis for grounding the statement, the intuitive meaning of which can be expressed as "I know that object x does not belong to category c."

Definition 5.15. Let the time point t and the extended state of cognitive processes $SPE(t)$ described by the episode $ExtEpisode(t)$ and the set of cognitive models M containing the well-defined model m_c be given. For each object $x \in X_t$ and category c we assume that the epistemic satisfaction relation $SPE(t) \vDash_G Know(x \notin c)$ holds if and only if

$$\forall r \in R \ \ f(r, o_c^\star) \geqslant \tau_c^-$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t. This definition is an extension of the definition 5.6 and can also be used for objects with complete information, since then $R = \{o\}$.

We consider that the boundary of category's model m_c includes objects that may or may not belong to category c. Therefore, including an object in the boundary of categories (lines 11 to 20) is the basis for establishing modal statements with operators of beliefs and possibilities, the intuitive meaning of which can be expressed as "I believe that object x belongs/does not belong to category c." and "I find it possible that object x belongs/does not belong to category c."

The statements with the knowledge operator were grounded only on the basis of the distance from the category prototype, which is equivalent to examining belonging to the appropriate area of the category model – the core or the outer region of the category. In the case of operators of belief and possibility, the distance to the prototype is not a sufficient basis for determining the degree of certainty of the verbal judgment regarding belonging to a category. Similar to chapter 5.2 for complete representation, to establish the degree of certainty we use the concept of the relative grounding strength λ_c from definition 5.8. The above coefficient is also computed for the epistemic neighborhood EN_c, but this time defined for a set of objects, rather than for a single object.

Definition 5.16. For a set of objects $R \subseteq O$ by epistemic neighborhood EN_c we understand a set of objects defined as follows:

$$EN_c(R, \varepsilon) = \bigcup_{r \in R} \{e \in (E_c^+(Exp_c) \cup E_c^-(Exp_c)) : f(e, r) \leqslant \varepsilon\}$$

where $\varepsilon \in \mathbb{R}$ is called the radius of the epistemic neighborhood.

Definition 5.17. Let the time point t, the extended state of cognitive processes $SPE(t)$ described by the episode $ExtEpisode(t)$ and the set of cognitive models M containing the well-defined model m_c, the radius of the epistemic neighborhood ε and $\lambda_{minBel} \in (0, 1]$ threshold be given. For any object $x \in X_t$ and category c we assume that epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \in c)$ and $SPE(t) \vDash_G Pos(x \notin c)$ hold if and only if

$$\left(\exists r \in R \;\; \tau_c^+ < f(r, o_c^\star) < \tau_c^-\right) \wedge \left(\lambda_c(EN_c(R, \varepsilon)) \geqslant \lambda_{minBel}\right)$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t. This definition is an extension of the definition 5.9 and can also be used for objects with complete information, since then $R = \{o\}$.

Definition 5.18. Let the time point t, the extended state of cognitive processes $SPE(t)$ described by the episode $ExtEpisode(t)$ and the set of cognitive models M containing the well-defined model m_c, the radius of the epistemic neighborhood ε and $\lambda_{minBel} \in (0, 1]$ threshold be given. For any object $x \in X_t$ and category c we assume that epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \notin c)$ and $SPE(t) \vDash_G Pos(x \in c)$ hold if and only if

$$\left(\exists r \in R \;\; \tau_c^+ < f(r, o_c^\star) < \tau_c^-\right) \wedge \left(\lambda_c(EN_c(R, \varepsilon)) < \lambda_{minBel}\right)$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t. This definition is an extension of the definition 5.10 and can also be used for objects with complete information, since then $R = \{o\}$.

5.3.1 Illustrative examples

Let us suppose that for model of category c the thresholds are equal to $\tau_c^+ = 5$ and $\tau_c^- = 8$. In $Episode(t_3)$ two objects $x_{3,8}$ and $x_{3,9}$ appeared in the agent's range of perception. Suppose that in the agent's cognition process the object $x_{3,8}$ from working memory corresponds to the object $o_8 \in ExtO$ in embodied ontology, and the object $x_{3,9}$ from working memory corresponds to the object $o_9 \in ExtO$ in embodied ontology.

Let us suppose that object o_8 does not have a specific value for one of its binary attributes. According to the algorithm 5.3 the statements are grounded on the basis of the set of objects $o \in O$ with all attribute values determined, obtained as a result of the *cReps* function. For the function defined as in the algorithm 5.2 the object o_8 with incomplete information corresponds to two objects with complete information.

Let us assume, that *cReps*$(o_8) = \{o_{11}, o_{12}\}$ and distances between the objects and the prototype are $f(o_{11}, o_c^\star) = 3$ and $f(o_{12}, o_c^\star) = 3$. Since both objects o_{11} and o_{12} are located in the core area, according to the definition 5.14 epistemic satisfaction relation holds for formula $Know(x_{3,8} \in c)$ and such a formula should be added to the set of statements S_t. The intuitive meaning of the formula can be expressed as "I know that object $x_{3,8}$ belongs to category c". The above situation is presented in the figure 5.4.

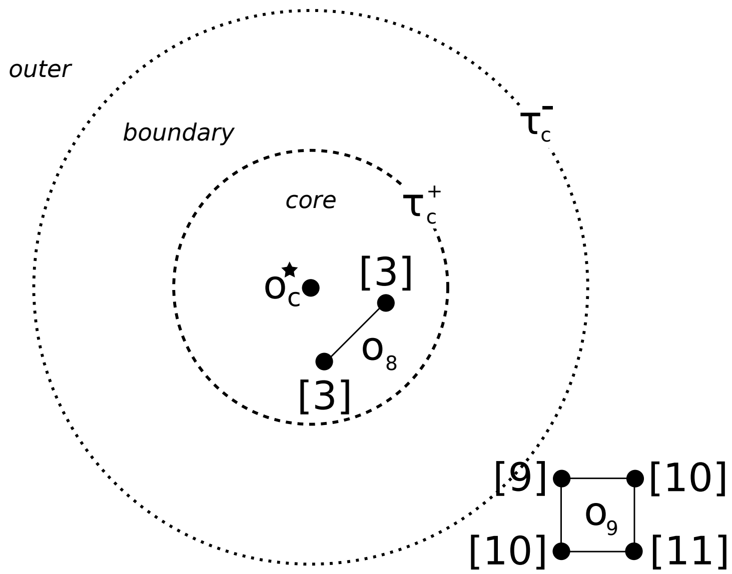


Figure 5.4: Grounding of objects with incomplete information in the core and the outer region of category model.

As for the object o_9, let us assume that it does not have a specific value for two binary attributes. For the function *cReps* defined as in the algorithm 5.2 the object o_9 with incomplete information corresponds to four objects with complete information. Let us assume, that *cReps*$(o_9) = \{o_{21}, o_{22}, o_{23}, o_{24}\}$ and distances between the objects and the prototype are $f(o_{21}, o_c^\star) = 9$, $f(o_{22}, o_c^\star) = 10$, $f(o_{23}, o_c^\star) = 10$ and $f(o_{24}, o_c^\star) = 11$. Since all objects $o_{21}...o_{24}$ are located in the outer region, according to the definition 5.15 epistemic satisfaction relation holds for formula $Know(x_{3,9} \notin c)$ and such a formula should be added to the set of statements S_t. The intuitive meaning of the formula can be expressed as "I know that object $x_{3,9}$ does not belong to category c". The above situation is also presented in the figure 5.4.

The figure 5.5 shows the situation, in which as before *cReps*$(o_9) = \{o_{21}, o_{22}, o_{23}, o_{24}\}$ but this time $f(o_{21}, o_c^\star) = 4$, $f(o_{22}, o_c^\star) = 7$, $f(o_{23}, o_c^\star) = 7$ and $f(o_{24}, o_c^\star) = 9$. Since the objects o_{22} and o_{23} are located in the boundary of category's model, to determine the correct formula, it is necessary to calculate the relative grounding strength for the set $R =$ *cReps*(o_9). The first step is to establish the radius of the epistemic neighborhood ε. As already mentioned, we apply the formula for the linear dependence of the radius on

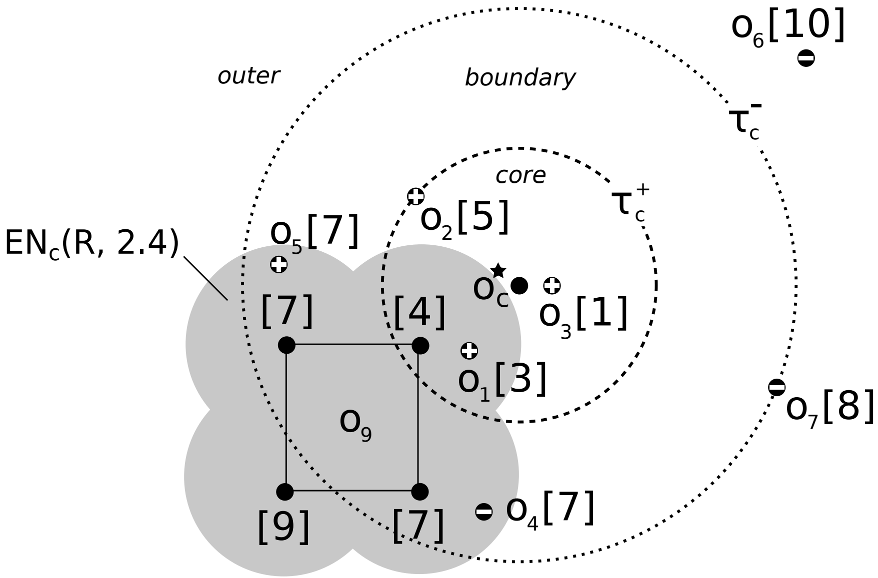


Figure 5.5: Grounding of objects with incomplete information in multiple regions of category model.

the width of the boundary. Assuming $\alpha = 0.8$ we get $\varepsilon = ER(\tau_c^-, \tau_c^+) = \alpha(\tau_c^- - \tau_c^+) = 0.8 \cdot (8 - 5) = 2.4$.

According to the definition 5.16 we calculate the epistemic neighborhood of the set $R = cReps(o_9)$, i.e. $EN_c(R, \varepsilon) = EN_c(\{o_{21}, o_{22}, o_{23}, o_{24}\}, 2.4)$. For this purpose, it is necessary to calculate the distance from the objects $o_{21}...o_{24}$ to the objects in the model of c located in embodied ontology. Let us assume that the above distances are given as in the table 5.4 and that:

- $E_c^+(Exp_c) = \{o_1, o_2, o_3, o_5\}$
- $E_c^-(Exp_c) = \{o_4, o_6, o_7\}$.

o_i	o_1	o_2	o_3	o_4	o_5	o_6	o_7
$f(o_{21}, o_i)$	1	3	3	4	4	14	10
$f(o_{22}, o_i)$	4	4	7	7	2	16	14
$f(o_{23}, o_i)$	3	5	7	2	7	16	12
$f(o_{24}, o_i)$	6	9	10	5	5	19	15

Table 5.4: Distances between objects in the example for incomplete representation.

Based on the above data, we determine $EN_c(R, \varepsilon) = EN_c(R, 2.4) = \{o_1, o_4, o_5\}$. According to the definition 5.8 we can calculate the relative grounding strength $\lambda_c(EN_c(R, 2.4))$ $= \frac{|\{o_1, o_5\}|}{|\{o_1, o_4, o_5\}|} = 2/3$.

Let us assume $\lambda_{minBel} = 0.5$. Such a threshold value means that if at least half of the elements in the epistemic neighborhood of the considered object are positive experiences, then the agent is willing to establish a statement with the operator of the belief that the object belongs to the category. On the other hand, if in the epistemic neighborhood of the considered object more than half of the elements are negative experiences, then the agent is willing to establish a statement with the operator of the possibility regarding the belonging of the object to the category.

Since $\lambda_c(EN_c(R, 2.4)) = 2/3 \geqslant \lambda_{minBel} = 0.5$ according to definition 5.17 epistemic satisfaction relation holds for formulas $Bel(x_{3,9} \in c)$ and $Pos(x_{3,9} \notin c)$ and such formulas should be added to the set of statements S_t. The intuitive meaning of the formulas can be expressed as "I believe that object $x_{3,9}$ belongs to category c." and "I find it possible that object $x_{3,9}$ does not belong to category c."

5.4 Interaction between teacher and agent

The presented model has an explicit knowledge representation, which allows to load ready-made structures representing categories into the agent's memory. However, in practice it makes more sense and is easier to use the strategy of learning the knowledge from examples, described in the chapter 4. This strategy assumes that the teacher labels objects in the agent's environment in a way, that allows the agent to gather the necessary knowledge. In practical applications, the number of objects in agent's perception might be quite large, so labeling all of them would be a tedious task. Which objects should be labeled can be determined if the teacher is able to identify which categories the agent has learned correctly and which are different from the conceptual framework used by the teacher. This is where statements generated by the agent can help. On their basis, the teacher can determine whether the agent has learned the category properly and attempt to correct the agent's improper linguistic behavior. Therefore, we are dealing with an interactive process, which is a kind of dialogue that allows also the agent to learn the concepts used by the teacher by interleaving the use of the learning strategy from the chapter 4 and the principles of statements grounding described in the chapter 5.

Situations that may arise in the above learning process are presented below, along with an explanation of how the agent's cognitive structures will adapt to the information newly obtained from the teacher. It is a demonstration of the flexibility of the model, allowing an interactive approach to be used in practice.

S1. The agent is more or less "convinced" that the object belongs to the category, while the teacher believes that the object cannot be assigned to the category. The teacher can therefore label the object with a negative label relating to the above category. The result should be an update of the agent's cognitive structures such that in the future the object is not assigned to a given category or, if this is not possible, introduces more uncertainty in the cognitive semantics related to a given category. Examples of this type of interaction will be presented in the following scenarios.

Scenario with negative example (a) Let us suppose, that there is the entity e_1 in the agent's range of perception. Let in $Episode(t_n)$ the entity e_1 be represented in agent's working memory as the object $x_{n,1}$, without any label assigned by the teacher. Let this object be a realization of the object o_1 from the mental space, that is located in the boundary of the model of category c, as shown in the left part of the figure 5.6. According to the cognitive semantics presented in previous chapters, in such a situation it is possible to ground statements with the *Bel* or *Pos* operator. Let us assume that due to the epistemic neighborhood of the object o_1, it is valid to ground the statement with the *Pos* operator. This causes the agent to generate a statement:

– I find it possible that object $x_{n,1}$ belongs to category c.

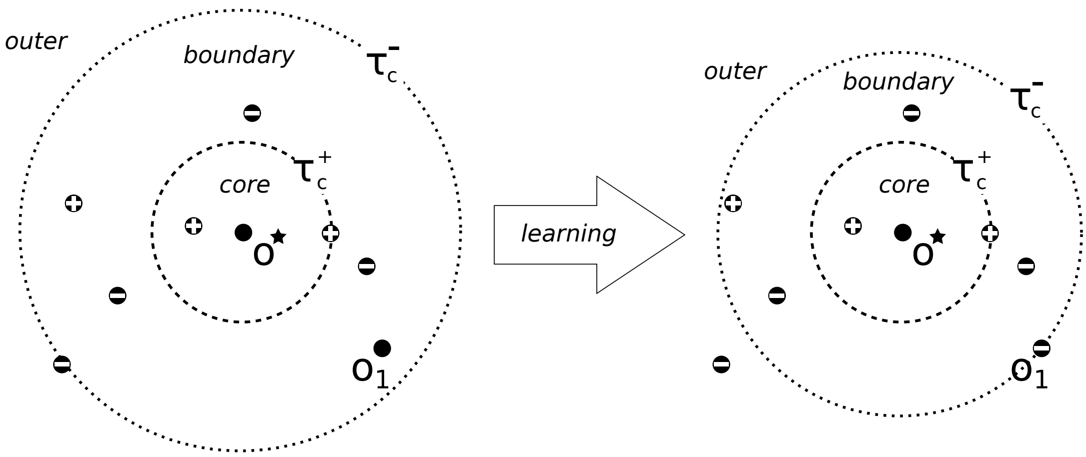


Figure 5.6: Updating model with negative example – scenario (a).

Suppose the teacher believes that the entity e_1 does not belong to the category c. Thus, the teacher tries to correct the agent's linguistic behavior by providing it with new information. Let the entity e_1 be represented in agent's working memory as the object $x_{n+1,1}$ in the next episode. The teacher assigns label *not-c* to the object $x_{n+1,1}$. Let us assume, that the values of the attributes describing the entity have not changed, so the object $x_{n+1,1}$ is also a realization of the object o_1 from mental space. Update of the model of category c reduces the radius τ_c^-, which means that the boundary area is also reduced, which can be seen in the right part of the figure 5.6. Since the object o_1 is now in the outer region, according to cognitive semantics presented in previous chapters, a statement should be grounded with the *Know* operator. Hence, the agent could generate the following statement:

– I know that object $x_{n+1,1}$ does not belong to category c.

Since this statement is consistent with the teacher's beliefs, they may finish the training of the agent in terms of category c. Let us note, that in the scenario above, updating the agent's knowledge led to a reduction in the uncertainty with respect to the category c.

Scenario with negative example (b) Let us suppose, that there is the entity e_1 in the agent's range of perception. Let in $Episode(t_n)$ the entity e_1 be represented in agent's working memory as the object $x_{n,1}$, without any label assigned by the teacher. Let this object be a realization of the object o_1 from the mental space, that is located in the core of the model of category c, as shown in the left part of the figure 5.7. According to the cognitive semantics presented in previous chapters, in such a situation it is valid to ground statements with the *Know* operator. This causes the agent to generate a statement:

– I know that object $x_{n,1}$ belongs to category c.

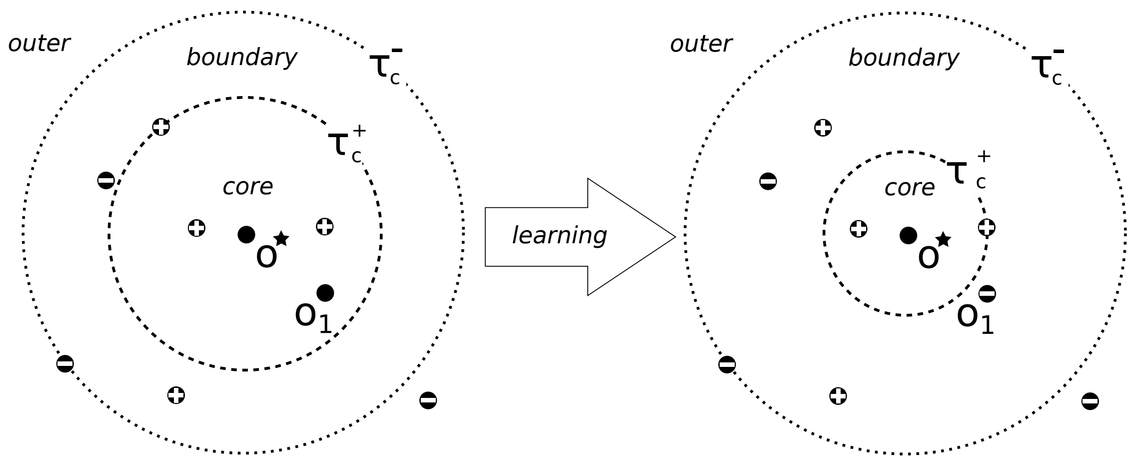


Figure 5.7: Updating model with negative example – scenario (b).

Suppose the teacher believes that the entity e_1 does not belong to the category c. Thus, the teacher tries to correct the agent's linguistic behavior by providing it with new information. Let the entity e_1 be represented in agent's working memory as the object $x_{n+1,1}$ in the next episode. The teacher assigns label *not-c* to the object $x_{n+1,1}$. Let us assume, that the values of the attributes describing the entity have not changed, so the object $x_{n+1,1}$ is also a realization of the object o_1 from mental space. Update of the model of category c reduces the radius τ_c^+, that is the core area is also reduced, which can be seen in the right part of the figure 5.7. Since the object o_1 is now in the boundary, according to cognitive semantics presented in previous chapters, a statement should be grounded with the Bel or Pos operator. Let us assume that due to the epistemic neighborhood of the object o_1, it is valid to ground the statement with the Bel operator. Hence, the agent could generate the following statement:

– I believe that object $x_{n+1,1}$ belongs to category c.

Such a statement is still not in line with the teacher's beliefs, although the agent's certainty that the object belongs to the c category has decreased. The teacher can continue training the agent in terms of category c by repeating the labeling of the e_1 entity or similar objects not belonging to the c category. This may lead to a gradual reduction in the agent's belief that the object belongs to the category c. However, in this case, it is not possible to achieve full agreement of the beliefs of the agent and the teacher, due to the wide range of boundary, where positive and negative examples are "mixed". This may be due to the reasons discussed in the previous chapters (different capabilities of agent and teacher perceptions, non-radial category structure, etc.) Let us note, that in the above scenario, updating the agent's knowledge led to an increased uncertainty with respect to the c category.

S2. The agent expresses the low level of possibility of belonging of the object to the category, which is consistent with the beliefs of the teacher. The teacher can label the object with a positive label relating to the above category. This should cause the update of the agent's cognitive structures, such that will strengthen the agent's belief that the object belongs to the category. This, in turn, will result in the fact that in the future the object with the same or similar values of attributes may be described by the agent with a statement expressing strong belief or even certainty regarding belonging to a category. Examples of this type of interaction will be presented in the following scenarios.

Scenario with positive example (a) Let us suppose, that there is the entity e_1 in the agent's range of perception. Let in $Episode(t_n)$ the entity e_1 be represented in agent's working memory as the object $x_{n,1}$, without any label assigned by the teacher. Let this object be a realization of the object o_1 from the mental space, that is located in the boundary of the model of category c, as shown in the left part of the figure 5.8. According to the cognitive semantics presented in previous chapters, in such a situation it is possible to ground statements with the Bel or Pos operator. Let us assume that due to the epistemic neighborhood of the object o_1, it is valid to ground the statement with the Bel operator. This causes the agent to generate a statement:

– I believe that object $x_{n,1}$ belongs to category c.

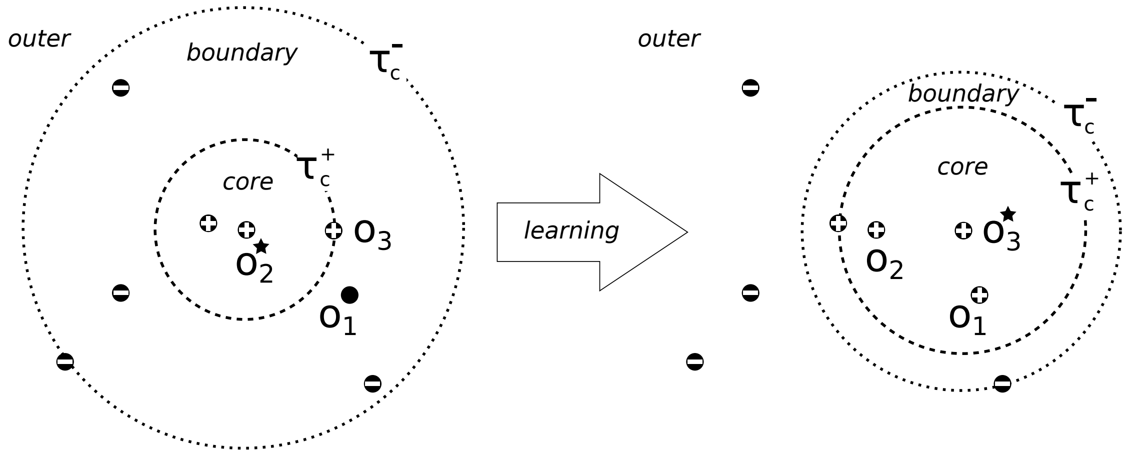


Figure 5.8: Updating model with positive example – scenario (a).

Suppose the teacher believes that the entity e_1 does belong to the category c. Based on the agent's statement, the teacher concludes that the agent is not completely certain that the entity belongs to the category c. Thus, the teacher tries to correct the agent's linguistic behavior by providing it with new information. Let the entity e_1 be represented in agent's working memory as the object $x_{n+1,1}$ in the next episode. The teacher assigns label *is-c* to the object $x_{n+1,1}$. Let us assume, that the values of the attributes describing the entity have not changed, so the object $x_{n+1,1}$ is also a realization of the object o_1 from mental space. Update of the model of category c causes the object o_3 to be selected as the prototype $o_c^\star$ of the c category model, instead of the o_2 as before. In addition, the radius τ_c^- is reduced and τ_c^+ increases, resulting in a significant reduction of the boundary area, which can be seen in the right part of the figure 5.8. Since the object o_1 is now in the core, according to cognitive semantics presented in previous chapters, a statement should be grounded with the $Know$ operator. Hence, the agent could generate the following statement:

– I know that object $x_{n+1,1}$ belongs to category c.

Since this statement is consistent with the teacher's beliefs, they may finish the training of the agent in terms of category c. Let us note, that in the scenario above, updating the agent's knowledge led to a reduction in the uncertainty with respect to the category c, thanks to the extension of core and outer region. The provision of new information by the teacher also shifted the "center of gravity" (the prototype) of the category to a more favorable point in the O universe.

Scenario with positive example (b) Let us suppose, that there is the entity e_1 in the agent's range of perception. Let in $Episode(t_n)$ the entity e_1 be represented in agent's working memory as the object $x_{n,1}$, without any label assigned by the teacher. Let this object be a realization of the object o_1 from the mental space, that is located in the outer region of the model of category c, as shown in the left part of the figure 5.9. According to the cognitive semantics presented in previous chapters, in such a situation it is valid to ground statements with the $Know$ operator. This causes the agent to generate a statement:

– I know that object $x_{n,1}$ does not belong to category c.

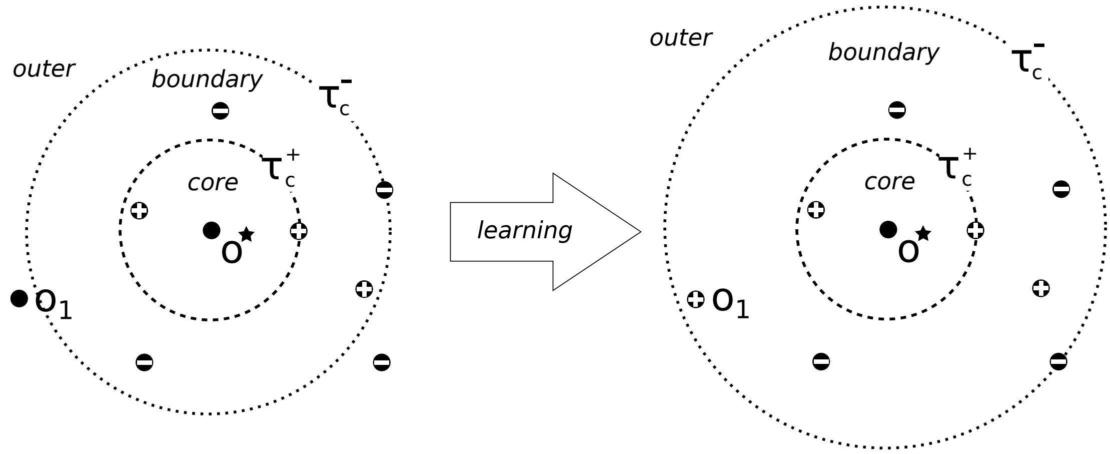


Figure 5.9: Updating model with positive example – scenario (b).

Suppose the teacher believes that the entity e_1 does belong to the category c. Thus, the teacher tries to correct the agent's linguistic behavior by providing it with new information. Let the entity e_1 be represented in agent's working memory as the object $x_{n+1,1}$ in the next episode. The teacher assigns label *is-c* to the object $x_{n+1,1}$. Let us assume, that the values of the attributes describing the entity have not changed, so the object $x_{n+1,1}$ is also a realization of the object o_1 from mental space. Update of the model of category c increases the radius τ_c^-, that is the outer region is also increased, which can be seen in the right part of the figure 5.9. Since the object o_1 is now in the boundary, according to cognitive semantics presented in previous chapters, a statement should be grounded with the Bel or Pos operator. Let us assume that due to the epistemic neighborhood of the object o_1, it is valid to ground the statement with the Pos operator. Hence, the agent could generate the following statement:

– I find it possible that object $x_{n+1,1}$ belongs to category c.

Such a statement is still not in line with the teacher's beliefs, although the agent now admits the possibility that the object belongs to the category c. The teacher can continue training the agent in terms of category c by repeating the labeling of the e_1 entity or similar objects belonging to the c category. This may lead to a gradual increase in the agent's belief that the object belongs to the category c. However, in this case, it is not possible to achieve full agreement of the beliefs of the agent and the teacher, due to the wide range of boundary, where positive and negative examples are "mixed". This may be due to the reasons discussed in the previous chapters (different capabilities of agent and teacher perceptions, non-radial category structure, etc.) Let us note, that in the above scenario, updating the agent's knowledge led to an increased uncertainty with respect to the c category.

S3. The agent does not express any opinion on the membership of the object to the category, to which the teacher would like to assign the object. Such a situation is possible when this category is unknown to the agent, because no assignment of an object to the above category has appeared in the agent's experience so far. The teacher can give the object a positive label relating to the category above. As a result, the agent should create a cognitive structure relevant to the new category.

As one can see, the agent's learning process is highly interactive. In order for it to work properly, at least in the initial phase, supervision by the teacher is necessary. After the teacher determines that all the agent's statements are rational, the learning process can be completed and the agent can work autonomously, generating statements about the objects in the environment.

5.5 Properties of semantics

The cognitive semantics of modal statements about the membership of an object to a category presented in this work has several properties, that are important for the proper conduct of a conversation with a human being. In common-sense interpretation of agent's statements, it is important that:

- it is not possible to utter certain statements about the same object simultaneously, e.g. it is not acceptable for the agent to generate the following statements in one episode, as they would be considered nonsensical/contradictory by other participants of communication:
  - "I know that object x belongs to category c",
  - "I know/I believe/I find it possible that object x does not belong to category c",

- it is possible to utter certain statements simultaneously regarding the same object, e.g. it is permissible for the agent to generate the following statements in one episode:
  - "I believe that object x belongs to category c",
  - "I find it possible that object x does not belong to category c".

Such properties are presented below, along with the proofs based on the definitions of epistemic satisfaction relations.

The theorems 5.1 and 5.2 concern a fairly obvious common-sense limitation, that the agent should not make statements indicating that it knows, that the object both belongs to and does not belong to the category c.

Theorem 5.1. *For the considered agent system, if relation $SPE(t) \vDash_G Know(x \in c)$ holds, then relation $SPE(t) \vDash_G Know(x \notin c)$ does not hold.*

Proof. The epistemic satisfaction relation $SPE(t) \vDash_G Know(x \in c)$ holds (definition 5.14) if and only if

$$\forall r \in R \;\; f(r, o^\star) \leqslant \tau_c^+$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t. In previous chapters we assumed that for a well-defined model always $\tau_c^+ < \tau_c^-$ (condition 4.1.1). It follows that

$$\forall r \in R \;\; f(r, o^\star) < \tau_c^-.$$

Thus, the condition $\forall r \in R \;\; f(r, o^\star) \geqslant \tau_c^-$ required for epistemic satisfaction relation $SPE(t) \vDash_G Know(x \notin c)$ is not fulfilled (definition 5.15). ■

Theorem 5.2. *For the considered agent system, if relation $SPE(t) \vDash_G Know(x \notin c)$ holds, then relation $SPE(t) \vDash_G Know(x \in c)$ does not hold.*

Proof. The epistemic satisfaction relation $SPE(t) \vDash_G Know(x \notin c)$ holds (definition 5.15) if and only if

$$\forall r \in R \;\; f(r, o^\star) \geqslant \tau_c^-$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t. In previous chapters we assumed that for a well-defined model always $\tau_c^+ < \tau_c^-$ (condition 4.1.1). It follows that

$$\forall r \in R \;\; f(r, o^\star) > \tau_c^+.$$

Thus, the condition $\forall r \in R \;\; f(r, o^\star) \leqslant \tau_c^+$ required for epistemic satisfaction relation $SPE(t) \vDash_G Know(x \in c)$ is not fulfilled (definition 5.14). ■

The next group of theorems (5.3 – 5.6) deals with situations where the agent generates a statement indicating that it knows that the object belongs to the category c or that it does not belong to the category c. In both cases, the agent should not produce simultanously statements with weaker confidence about the class-membership of the object.

Theorem 5.3. *For the considered agent system, if relation* $SPE(t) \vDash_G Know(x \in c)$ *holds, then*

- *relation* $SPE(t) \vDash_G Bel(x \in c)$ *does not hold,*
- *relation* $SPE(t) \vDash_G Bel(x \notin c)$ *does not hold,*
- *relation* $SPE(t) \vDash_G Pos(x \in c)$ *does not hold,*
- *relation* $SPE(t) \vDash_G Pos(x \notin c)$ *does not hold.*

Proof. The epistemic satisfaction relation $SPE(t) \vDash_G Know(x \in c)$ holds (definition 5.14) if and only if

$$\forall r \in R \;\; f(r, o^\star) \leqslant \tau_c^+$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t.

Thus, the condition $\exists r \in R \;\; \tau_c^+ < f(r, o^\star)$ required for epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \in c)$ and $SPE(t) \vDash_G Pos(x \notin c)$ is not fulfilled (definition 5.17). The same condition is required for epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \notin c)$ and $SPE(t) \vDash_G Pos(x \in c)$ (definition 5.18). ■

Theorem 5.4. *For the considered agent system, if any of the following relationships hold*

- $SPE(t) \vDash_G Bel(x \in c)$,
- $SPE(t) \vDash_G Bel(x \notin c)$,
- $SPE(t) \vDash_G Pos(x \in c)$,
- $SPE(t) \vDash_G Pos(x \notin c)$,

then relation $SPE(t) \vDash_G Know(x \in c)$ *does not hold.*

Proof. This theorem is the contraposition of theorem 5.3. ■

Theorem 5.5. *For the considered agent system, if relation $SPE(t) \vDash_G Know(x \notin c)$ holds, then*

- *relation $SPE(t) \vDash_G Bel(x \in c)$ does not hold,*
- *relation $SPE(t) \vDash_G Bel(x \notin c)$ does not hold,*
- *relation $SPE(t) \vDash_G Pos(x \in c)$ does not hold,*
- *relation $SPE(t) \vDash_G Pos(x \notin c)$ does not hold.*

Proof. The epistemic satisfaction relation $SPE(t) \vDash_G Know(x \notin c)$ holds (definition 5.15) if and only if

$$\forall r \in R \;\; f(r, o^{\star}) \geqslant \tau_c^-$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information and x is observed realization of mental object o in time point t.

Thus, the condition $\exists r \in R \;\; f(r, o^{\star}) < \tau_c^-$ required for epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \in c)$ and $SPE(t) \vDash_G Pos(x \notin c)$ is not fulfilled (definition 5.17). The same condition is required for epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \notin c)$ and $SPE(t) \vDash_G Pos(x \in c)$ (definition 5.18). ■

Theorem 5.6. *For the considered agent system, if any of the following relationships hold*

- $SPE(t) \vDash_G Bel(x \in c)$,
- $SPE(t) \vDash_G Bel(x \notin c)$,
- $SPE(t) \vDash_G Pos(x \in c)$,
- $SPE(t) \vDash_G Pos(x \notin c)$,

then relation $SPE(t) \vDash_G Know(x \notin c)$ does not hold.

Proof. This theorem is the contraposition of theorem 5.5. ■

If the agent generates the statement indicating, that it believes that the object belongs (or does not belong) to the category c, then it should not produce at the same time the statement with weaker confidence. Neither should it produce the statement indicating, that it believes the opposite state of membership. The correct behavior of the agent in above situations is proven for theorems 5.7 and 5.8.

Theorem 5.7. *For the considered agent system, if relation $SPE(t) \vDash_G Bel(x \in c)$ holds, then*

- *relation* $SPE(t) \vDash_G Bel(x \notin c)$ *does not hold,*
- *relation* $SPE(t) \vDash_G Pos(x \in c)$ *does not hold.*

Proof. The epistemic satisfaction relation $SPE(t) \vDash_G Bel(x \in c)$ holds (definition 5.17) if and only if

$$\left(\exists r \in R \;\; \tau_c^+ < f(r, o^\star) < \tau_c^-\right) \wedge \left(\lambda_c(EN_c(R, \varepsilon)) \geqslant \lambda_{minBel}\right)$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information, x is observed realization of mental object o in time point t and $EN_c(R, \varepsilon)$ is epistemic neighborhood of set R with radius ε.

Thus, the condition $\lambda_c(EN_c(R, \varepsilon)) < \lambda_{minBel}$ required for epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \notin c)$ and $SPE(t) \vDash_G Pos(x \in c)$ is not fulfilled (definition 5.18). ∎

Theorem 5.8. *For the considered agent system, if relation* $SPE(t) \vDash_G Bel(x \notin c)$ *holds, then*

- *relation* $SPE(t) \vDash_G Bel(x \in c)$ *does not hold,*
- *relation* $SPE(t) \vDash_G Pos(x \notin c)$ *does not hold.*

Proof. The epistemic satisfaction relation $SPE(t) \vDash_G Bel(x \notin c)$ holds (definition 5.18) if and only if

$$\left(\exists r \in R \;\; \tau_c^+ < f(r, o^\star) < \tau_c^-\right) \wedge \left(\lambda_c(EN_c(R, \varepsilon)) < \lambda_{minBel}\right)$$

where R is a set of objects with complete information corresponding to the object o with (possibly) incomplete information, x is observed realization of mental object o in time point t and $EN_c(R, \varepsilon)$ is epistemic neighborhood of set R with radius ε.

Thus, the condition $\lambda_c(EN_c(R, \varepsilon)) \geqslant \lambda_{minBel}$ required for epistemic satisfaction relations $SPE(t) \vDash_G Bel(x \in c)$ and $SPE(t) \vDash_G Pos(x \notin c)$ is not fulfilled (definition 5.17). ∎

Analogically to the above, if the agent generates the statement indicating, that it is possible that the object belongs (or does not belong) to the category c, then it should not produce at the same time the statement with stronger confidence – theorems 5.9 and 5.10.

Theorem 5.9. *For the considered agent system, if relation* $SPE(t) \vDash_G Pos(x \in c)$ *holds, then relation* $SPE(t) \vDash_G Bel(x \in c)$ *does not hold.*

Proof. This theorem is the contraposition of second part of theorem 5.7. ∎

Theorem 5.10. *For the considered agent system, if relation $SPE(t) \vDash_G Pos(x \notin c)$ holds, then relation $SPE(t) \vDash_G Bel(x \notin c)$ does not hold.*

Proof. This theorem is the contraposition of second part of theorem 5.8. ∎

If the agent expresses belief that the object belongs to some category c, it is rational that the agent accepts the possibility that this object does not belong to the category c. So it should be possible for the agent to express both of above opinions at the same state of knowledge. Possibly, such pairs of statements could be connected with additional language connector (not defined formally in this work), e.g. "I believe that object x belongs to category c, however I find it possible that object x does not belong to category c".

Similarly, if the agent expresses belief that the object does not belong to some category c, it is rational that the agent accepts possibility that this object does belong to the category c. The next two theorems (5.11 and 5.12) are proven for the above situations.

Theorem 5.11. *For the considered agent system, relations $SPE(t) \vDash_G Bel(x \in c)$ and $SPE(t) \vDash_G Pos(x \notin c)$ hold in the same state of knowledge.*

Proof. It follows directly from definition 5.17, where the conditions of epistemic satisfaction relation are the same for both formulas. ∎

Theorem 5.12. *For the considered agent system, relations $SPE(t) \vDash_G Bel(x \notin c)$ and $SPE(t) \vDash_G Pos(x \in c)$ hold in the same state of knowledge.*

Proof. It follows directly from definition 5.18, where the conditions of epistemic satisfaction relation are the same for both formulas. ∎

5.6 Connection with Basic Grounding Model

In this chapter, formal definitions from the Basic Grounding Model [31, 32], presented initially in the chapter 3.3, will be given, and the relationship between BGM and the model described in the dissertation will be analyzed. We will refer to the latter model as the Category Grounding Model (CGM) for convenience.

The system described by the BGM model is equipped with a mechanism/ability to correctly identify individual objects, i.e. it observes the entity and knows that it is the individual entity e. The consequence on the formal level is the possibility of marking the real entity and its internal image in the system with the same symbol e – each observed object is "transferred" one-to-one to the agent's mental space. The main difference between the models is therefore the fact that in the BGM model there is no distinction between the real entities observed by the agent and objects in the agent's mental space, while in the CGM model there is such a distinction. Thus, in the BGM model it is not possible for two entities with the same values of attributes to be mapped to one object in mental space, as is the case in the CGM model. The differences between the models in this respect are shown in the figure 5.10.

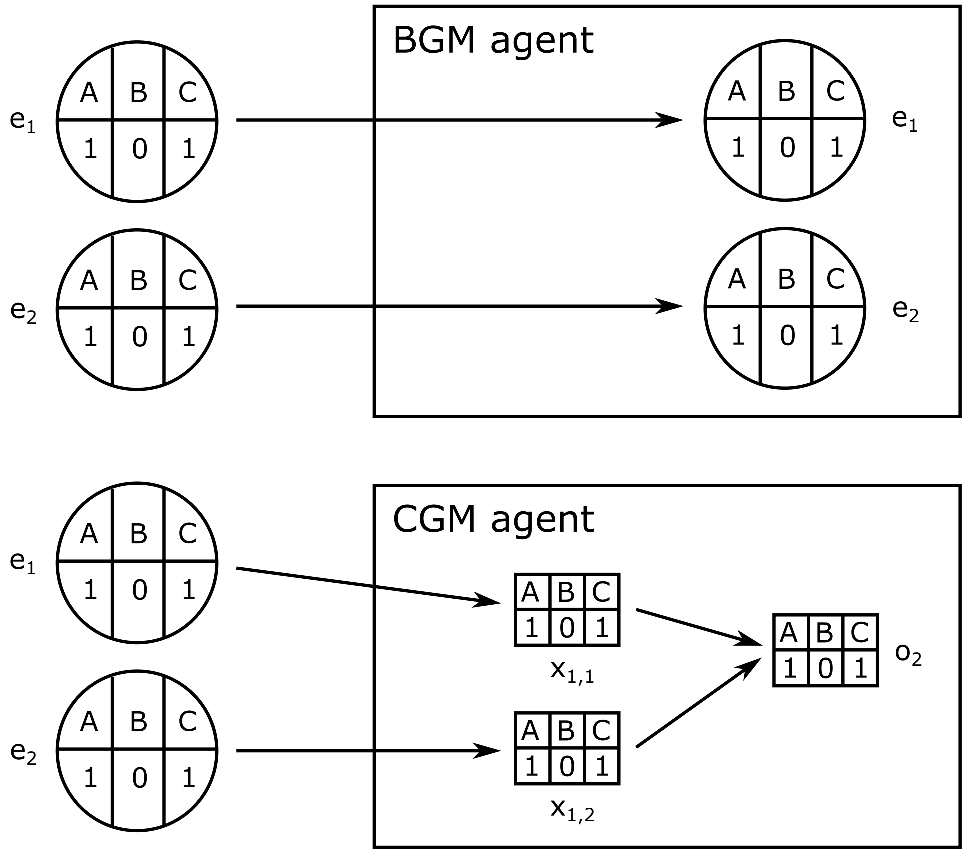


Figure 5.10: Difference between BGM and CGM models.

Thus, in the CGM model, for application reasons, the internal model is significantly expanded, considering the transition: the entity → the object in the perception of the agent → the object in the agent's mental space. To obtain an unambiguous interpretation, it was necessary to introduce more symbols in the CGM model, corresponding to the internal and external representation of the objects. The above complication of the model is needed to obtain behaviors closer to the natural ones in the context of generating statements about the membership of the object to the category. The system becomes more complex but includes more functionality as there are fewer restrictions placed on it. In the BGM model, the agent must track entities in subsequent episodes, i.e. unambiguously recognize a given entity e_i in all experienced episodes, throughout the entire period of the agent's "life". In CGM, the continuity in object identification is not required.

For the above reasons, it was necessary to change the original notation of the BGM model. In publications on BGM, the set of objects O contains the set of real objects (entities) observed by the agent, while in the CGM model the set of objects O contains the universe of all objects that the agent can “imagine”. To avoid a name conflict, we will call a set of BGM objects a set of entities and denote them with the letter E. The set of objects O will only refer to the concept from the CGM model.

The basic definitions based on [31] necessary to formally define the grounding relationship in the meaning of the BGM are presented below. In BGM the external environment can be defined as a dynamic system of objectively existing atom entities. Particular states of these entities are related to time points $T = \{t_0, t_1, t_2, ...\}$ ordered by the relation $\leqslant^{TM}$.

Definition 5.19. At each time point t the following tuple of sets is given

$$PS(t) =< E, P_1(t), ..., P_K(t) >$$

which elements are assigned the following roles and interpretations: $E = \{e_1, e_2, ..., e_M\}$ denotes the set of all atom entities located in the external world. Members of the set $\Delta = \{P_1, P_2, ..., P_K\}$ are the names of properties that can be attributed to entities from E. In particular, each entity $e \in E$ may or may not exhibit each property $P \in \Delta$.

Definition 5.20. Each internal representation of the result of observation carried out by the cognitive agent at a time point t is given as the tuple of sets, called the base profile:

$$BP(t) =< E, P_1^+(t), P_1^-(t), ..., P_K^+(t), P_K^-(t) >$$

where

- for each $j = 1, 2, ..., K$, the following relations $P_j^+(t) \subseteq E$ and $P_j^-(t) \subseteq E$ hold,
- for each $e \in E$ the relation $e \in P_j^+(t)$ holds if and only if the cognitive agent responsible for the creation of base profile $BP(t)$ perceived the entity e as exhibiting the property P_j at the time point t,
- for each $e \in E$ the relation $e \in P_j^-(t)$ holds if and only if the cognitive agent responsible for the creation of base profile $BP(t)$ did not perceive the entity e as exhibiting the property P_j at the time point t,
- for each $j = 1, 2, ..., K$ the condition $P_j^+(t) \cap P_j^-(t) = \varnothing$ holds; it represents the basic constraints of natural cognition, according to which it is not possible to perceive the same entity e as exhibiting and not exhibiting the same property P_j at the same time point t.

Definition 5.21. At each time point $t \in T$ the state of empirical knowledge about the external world is defined by a temporal collection of base profiles given as follows:

$$KS(t) = \{BP(t_n) : t_n \in T \quad and \quad t_n \leqslant^{TM} t\}.$$

Definition 5.22. Let the symbols $A^1(t)$ and $A^2(t)$ denote the so called grounding sets. The content of $A^1(t)$ and $A^2(t)$ is defined as follows:

$$A^1(t) = \{BP(t_n) : t_n \leqslant^{TM} t \quad and \quad BP(t_n) \in KS(t) \quad and \quad e \in P^+(t_n)\},$$

$$A^2(t) = \{BP(t_n) : t_n \leqslant^{TM} t \quad and \quad BP(t_n) \in KS(t) \quad and \quad e \in P^-(t_n)\}.$$

The set $A^1(t)$ consists of all base profiles from $KS(t)$, which have been created by the cognitive agent up to the time point t and represent the empirical experience of the entity e exhibiting the property P. The set $A^2(t)$ consists of all base profiles in which the entity e is represented as not exhibiting the property P.

Definition 5.23. The inductive strength of the grounding sets $A^i(t), i \in \{1, 2\}$, is given as its cardinality $G(A^i(t)) = card(A^i(t))$.

Definition 5.24. For the formulas $p(e)$ and $\neg p(e)$ the relative grounding strength λ_{BG} is given for each time point t by the following equations:

$$\lambda_{BG}(t, p(e)) = \frac{G(A^1(t))}{G(A^1(t)) + G(A^2(t))}$$

$$\lambda_{BG}(t, \neg p(e)) = \frac{G(A^2(t))}{G(A^1(t)) + G(A^2(t))}$$

The relative grounding strength will be compared with the modality thresholds. The following dependencies are assumed, which guarantee the basic rationality of linguistic behavior:

$$0 < \lambda_{minPos} < \lambda_{maxPos} \leqslant \lambda_{minBel} < \lambda_{maxBel} \leqslant 1.$$

Another dimension of cognition that influences the definition of epistemic satisfaction relation is the above mentioned distribution of the content of $A^1(t)$ and $A^2(t)$ over the conscious subspace $WS(t)$ and non-conscious cognitive subspace $LS(t)$. $WS(t)$ can be considered as a working memory and $LS(t)$ as a long term memory.

Definition 5.25. At each time point $t \in T$ the t-related state of cognition is described by the following binary partition of $KS(t)$:

$$PS(t) = \{WS(t), LS(t)\}$$

where the conditions $WS(t) \cup LS(t) = PS(t)$ and $WS(t) \cap LS(t) = \varnothing$ hold.

Definition 5.26. Let the t-related distribution of the empirical material of the formulas $p(e)$ and $\neg p(e)$, respectively, be given by the following set $PS(t) = \{WA^1(t), LA^1(t), WA^2(t), LA^2(t)\}$, where for i=1,2,

- $WA^i(t) = WS(t) \cap A^i(t)$,
- $LA^i(t) = LS(t) \cap A^i(t)$,
- $WA^i(t) \cap LA^i(t) = \varnothing$,
- $WA^i(t) \cup LA^i(t) = A^i(t)$.

Based on the above definitions, we can define the conditions necessary to ground specific formulas. They use the relation of epistemic satisfaction of the formula, just like the CGM model.

formula	**assigned meaning**
$p(e)$	Entity e exhibits property P.
$\neg p(e)$	Entity e does not exhibit the property P.
$Pos(\varphi)$	It is possible that φ.
$Bel(\varphi)$	I believe that φ.
$Know(\varphi)$	I know that φ.

Table 5.5: Commonsense semantics of atomic formulas in BGM model.

The main difference in syntax is that the formulas in the BGM model refer to the current state of the feature P in the entity e, while in the CGM model they express the belonging (or not) of the object x to the category c. Direct observation of the state of a feature in an object is thus replaced by the mechanism of placing the image of the observed object in relation to the internally (cognitively) built prototype related to the feature analyzed at the moment, i.e. the occurrence of which means belonging to the category, and the lack of occurrence as the opposite situation.

The membership to a given category can therefore be considered a feature of the object – e.g. belonging to the $Fruit$ category can be modeled as the $IsFruit$ attribute with the values Yes/No. However, it is not possible to apply directly to the CGM model the cognitive semantics described for the BGM model[31, 32]. Hence, it was necessary to develop the original cognitive semantics for the CGM model, as described in this dissertation. The definitions of epistemic satisfaction relations in both models are compared below.

Definition 5.27. Let a time point $t \in T$ and a distribution of base profiles $PS(t) = \{WA^1(t), LA^1(t), WA^2(t), LA^2(t)\}$ be given.

For each property $P \in \{P_1, ..., P_K\}$ and each entity $e \in E$ the epistemic satisfaction relation $PS(t) \vDash_{BG} Know(p(e))$ and $PS(t) \vDash_{BG} p(e)$ holds if and only if either the relation $e \in P^+(t)$ or the relations $e \in E \setminus (P^+(t) \cup P^-(t))$, $WA^1(t) \neq \varnothing$ and $\lambda_{BG}(t, p(e)) = 1$ hold.

As one can see, the conditions for grounding the non-modal formula $p(e)$ and the modal formula $Know(p(e))$ are identical. This is in line with common-sense principles governing the generation of the modal statements. Namely, the statement "The entity e exhibits the property P" is pronounced if and only if it is possible to equivalently express the phrase "I know that the entity e exhibits the property P". In the CGM model, non-modal statements are not considered, but of course it can be extended in a similar way. In further analysis, however, we will limit ourselves to modal formulas.

The definition distinguishes two groups of cognitive states in which the conditions for formula $Know(p(e))$ are satisfied. The first group of cognitive states includes states in which the current baseline profile (representing the last observation) asserts that the entity e exhibits the property P. Then it is not necessary to refer to previous experience in the exhibition of the property P by the entity e. The intensity of the latest experience guarantees the compliance of the constructed image of the environment with the state of external reality. Consequently, the formulas are grounded in the state of knowing the state of the property P in the entity e. For the CGM model, in the definitions 5.5 and 5.14 of the epistemic satisfaction relation of the formula $Know(x \in c)$ there is no equivalent of this cognitive state, because we consider belonging to a category to be something beyond the current value of a particular property or attribute. In many cases, the membership to a category extends over the entire lifetime of the object (e.g. the categories *Bird*, *Table*). There are also situations when the object can change its category membership (e.g. the categories *Dangerous*, *Beautiful*) but rather, it is likely that these changes span through more than one episode. For this reason, the definitions of the epistemic satisfaction relations in the CGM model refer to objects in the universe of the agent's mental space, and not to the observed entities. Introducing such an equivalent to the CGM model would require reference to the teacher's current labeling. However, it does not seem appropriate to relate the satisfiability of a relation simultaneously to the current observation of a real, concrete entity and object from the mental space, which is an idealized image of a certain being. This would also result in the agent behaving differently in situations where the entity is labeled in a given episode and where there is no such label. This would make it difficult for the teacher to judge the degree of the agent's mastery of the meaning of the category.

The second group of states includes ones in which it is not possible to know the current state of the property P in the entity e by observing the above entity. The agent must refer to previous empirical experience of the property P in the entity e. The conditions $WA^1(t) \neq \varnothing$ and $\lambda_{BG}(t, p(e)) = 1$ mean that $A^2(t) = \varnothing$, thus the agent never observed the property P not present in the entity e. The current view of the world thus supports its view of the unconditional presence of the property P in the entity e. The status of this type of knowledge is different than for the evident presence of P in the entity e due to its relativization to the recognized past states. However, the experienced pattern of the outside world offers no alternative. The agent simply knows, because it has not experienced a different state of the world. In the case of the CGM model, the formula $Know(x \in c)$ can be grounded, when the object x belongs to the core of the model of category c. This means that all objects as close or closer to the prototype of the category c, belong to c. As one can see, this is a broader definition than in the BGM model, in the sense that it refers to the agent's experience with many objects, not only to the current entity under consideration. Nevertheless, the principle is similar – the agent has not experienced a situation in which the object not belonging to the c category was closer to the prototype than the object x for which the relation of epistemic satisfaction of the formula $Know(x \in c)$ is being determined. Therefore, the situations in both models are functionally similar.

Definition 5.28. Let a time point $t \in T$ and a distribution of base profiles $PS(t) = \{WA^1(t), LA^1(t), WA^2(t), LA^2(t)\}$ be given.

For each property $P \in \{P_1, ..., P_K\}$ and each entity $e \in E$ the epistemic satisfaction relation $PS(t) \vDash_{BG} Know(\neg p(e))$ and $PS(t) \vDash_{BG} \neg p(e)$ hold if and only if either the relation $e \in P^-(t)$ or the relations $e \in E \setminus (P^+(t) \cup P^-(t))$, $WA^2(t) \neq \varnothing$ and $\lambda_{BG}(t, \neg p(e)) = 1$ hold.

Similar reasoning can be carried out for the relation of epistemic satisfaction of formulas $PS(t) \vDash_{BG} Know(\neg p(e))$ and $PS(t) \vDash_{BG} \neg p(e)$. This time, in the CGM model, the formula $Know(x \notin c)$ can be grounded, when the object x belongs to the outer region of the category c. This means that the agent did not experience a situation, where the object belonging to the category c was further from the prototype than the object x for which the epistemic satisfaction relation of the formula $Know(x \notin c)$ is determined. In the BGM model, this means that the agent has experienced only situations when the property P is not present in the entity e (meaning $A^1(t) = \varnothing$).

Definition 5.29. Let a time point $t \in T$, a distribution of base profiles $PS(t) = \{WA^1(t), LA^1(t), WA^2(t), LA^2(t)\}$ and a system of modality thresholds with the constraint $0 < \lambda_{minPos} < \lambda_{maxPos} \leqslant \lambda_{minBel} < \lambda_{maxBel} \leqslant 1$ be given.

For each property $P \in \{P_1, ..., P_K\}$ and each entity $e \in E$ the epistemic satisfaction relation $PS(t) \vDash_{BG} Bel(p(e))$ holds if and only if the relations $e \in E \setminus (P^+(t) \cup P^-(t))$, $WA^1(t) \neq \varnothing$ and $\lambda_{minBel} \leqslant \lambda_{BG}(t, p(e)) \leqslant \lambda_{maxBel}$ hold.

The formula $Bel(p(e))$ is epistemically satisfied when the state of the agent's cognitive processes does not allow for an unambiguous empirical determination of the state of the property P in the entity e. Such a situation occurs when the state of the property P in the entity e was not included in the last (most recent) environmental observation. Hence, it is expected to occur $e \in E \setminus (P^+(t) \cup P^-(t))$. Consequently, recognition about the state of the property P in the entity e is possible only by referring to the empirical experience to date. Including all other experiences related to the property P in the entity e (including complementary ones) is guaranteed by the previously adopted definition of the relative grounding strength $\lambda_{BG}(t, p(e))$. If the strength of the experience associated with the occurrence of the property P turns out to be large enough (i.e. the condition $\lambda_{minBel} \leqslant \lambda_{BG}(t, p(e)) \leqslant \lambda_{maxBel}$ is met), the state of processing the content of the property P in the entity e can be communicated to another conversation participant by sending a message of the form $Bel(p(e))$. In the case of the CGM model, the relative grounding strength λ_c is also used (definition 5.8), but again it refers to many previously observed objects, instead of only the current entity as in BGM. Also in CGM, grounding the formula $Bel(x \in c)$ is possible only when the condition $\lambda_{minBel} \leqslant \lambda_c(EN_c(o, \varepsilon))$ is met. In CGM the threshold λ_{maxBel} is not used, because the *Know* and *Bel* operators are chosen basing on the area in which the object x is located, and not on the basis of λ_c. In the CGM model, grounding the formula $Bel(x \in c)$ is equivalent to grounding the formula $Pos(x \notin c)$, which is treated as complementary.

Definition 5.30. Let a time point $t \in T$, a distribution of base profiles $PS(t) = \{WA^1(t), LA^1(t), WA^2(t), LA^2(t)\}$ and a system of modality thresholds with the constraint $0 < \lambda_{minPos} < \lambda_{maxPos} \leqslant \lambda_{minBel} < \lambda_{maxBel} \leqslant 1$ be given.

For each property $P \in \{P_1, ..., P_K\}$ and each entity $e \in E$ the epistemic satisfaction relation $PS(t) \vDash_{BG} Pos(p(e))$ holds if and only if the relations $e \in E \setminus (P^+(t) \cup P^-(t))$, $WA^1(t) \neq \varnothing$ and $\lambda_{minPos} \leqslant \lambda_{BG}(t, p(e)) \leqslant \lambda_{maxPos}$ hold.

The epistemic satisfaction of the above formula can be interpreted in a very similar way as the formula $Bel(p(e))$. Its definition only uses a different set of thresholds: λ_{minPos} and λ_{maxPos}. In the CGM model, the threshold λ_{minPos} is not needed, because the possibility of belonging to a category is excluded on the basis of assigning the object to the outer region of the category. Instead of the threshold λ_{maxPos} the condition $\lambda_c(EN_c(o, \varepsilon)) < \lambda_{minBel}$ is used, which corresponds to the assumption that $\lambda_{maxPos} = \lambda_{minBel}$. The fulfillment of the above condition means that it is justified to ground the formulas $Pos(x \in c)$ and $Bel(x \notin c)$.

Definition 5.31. Let a time point $t \in T$, a distribution of base profiles $PS(t) = \{WA^1(t), LA^1(t), WA^2(t), LA^2(t)\}$ and a system of modality thresholds with the constraint $0 < \lambda_{minPos} < \lambda_{maxPos} \leqslant \lambda_{minBel} < \lambda_{maxBel} \leqslant 1$ be given.

For each property $P \in \{P_1, ..., P_K\}$ and each entity $e \in E$ the epistemic satisfaction relation $PS(t) \vDash_{BG} Bel(\neg p(e))$ holds if and only if the relations $e \in E \setminus (P^+(t) \cup P^-(t))$, $WA^2(t) \neq \varnothing$ and $\lambda_{minBel} \leqslant \lambda_{BG}(t, \neg p(e)) \leqslant \lambda_{maxBel}$ hold.

Definition 5.32. Let a time point $t \in T$, a distribution of base profiles $PS(t) = \{WA^1(t), LA^1(t), WA^2(t), LA^2(t)\}$ and a system of modality thresholds with the constraint $0 < \lambda_{minPos} < \lambda_{maxPos} \leqslant \lambda_{minBel} < \lambda_{maxBel} \leqslant 1$ be given.

For each property $P \in \{P_1, ..., P_K\}$ and each entity $e \in E$ the epistemic satisfaction relation $PS(t) \models_{BG} Pos(\neg p(e))$ holds if and only if the relations $e \in E \backslash (P^+(t) \cup P^-(t))$, $WA^2(t) \neq \varnothing$ and $\lambda_{minPos} \leqslant \lambda_{BG}(t, \neg p(e)) \leqslant \lambda_{maxPos}$ hold.

The grounding of formulas indicating not belonging to the category has already been discussed above. In the BGM model, the satisfaction of the epistemic relation for analogous formulas is defined using a function $\lambda_{BG}(t, \neg p(e))$. Because it is complementary to $\lambda_{BG}(t, p(e))$ (in the sense that $\lambda_{BG}(t, p(e)) + \lambda_{BG}(t, \neg p(e)) = 1$), its interpretation is analogous to that for formulas $Bel(p(e))$ and $Pos(p(e))$.

Chapter 6

Implementation models

6.1 Introduction

6.1.1 Introductory remarks on possible implementations

Previous chapters present the general strategy of model and statement construction. Its implementation requires specifying the particular macrostructure and mechanism of candidate extraction. Their selection depends on the (known or assumed) structure of the categories that we want the agent to learn. This chapter provides examples of the use of three different macrostructures and their respective selection mechanisms.

Referring again to the psycholinguistic research[38] – the prototypical elements are the ones that:

(a) have the most properties in common with other elements of the category,

(b) and at the same time the least common properties with the elements of opposite categories.

In this chapter, three models are presented. The first of them is based on simple (unweighted) Hamming distance measure and it satisfies only the condition (a), yet it is easy to implement. The next two models are more complicated, based on Hamming distance with weights, and they take into account both prototype conditions.

One can consider as the macrostructure many other measures described in literature, including similarity[87] measures instead of distance functions, e.g. [28, 46, 51, 70].

6.1.2 Central element in applications – k-means and k-medoids clustering algorithms

A task of searching for candidates for a prototype is close to problem of choosing a central element for a cluster. There is a lot of different algorithms for clustering in literature, since there are many definitions of a cluster itself [13].

Calculation of central elements of a set and, in particular, a calculation of centroids and medoids constitutes a foundation of k-means[25, 26] and k-medoids[35] algorithms, as it is performed in each cycle of those clustering algorithms. Depending on a particular case, evaluations of centroids/medoids are performed in literature for a variety of distance functions. There is a lot of existing research focusing on optimization of the computational complexity of k-means and k-medoids algorithms. However, most of the approaches optimizing complexity of these algorithms focus strictly on aspects related to those algorithms themselves. That is, they usually aim to reduce the complexity by restricting the number of needed recalculations between the particular cycles of the main algorithm loop[30, 65, 82]. That provides results useful in a general case as they are independent of an underlying distance function. Moreover, the literature dealing with weighted distance functions assign weights only to objects, while in case of Weighted Hamming distance introduced in next subchapters, both weights of objects and weights of attributes are considered.

In the following examples involving Hamming distance (and some of its modifications) we will focus on showing how to effectively evaluate a centroid and a medoid of a multiset while optimizing according to an assumed distance function.

6.2 Hamming distance

6.2.1 Macrostructure

In this model, the macrostructure influences the selection of prototype candidates. For this reason, the selected function should allow the prototyping conditions given in the chapter 6.1.1 to be met. The condition (a) given there can be interpreted as meaning that the higher the number of common attributes between the elements of a category, the greater the degree of similarity and lower the distance between these elements. Hence, it is proposed to use the Hamming distance, i.e. the number of attributes which values for the o_i and o_j objects are different. [1]

Definition 6.1. The Hamming distance f_H for the vectors o_i and o_j is defined as

[1] The above distance measure is also called Manhattan or city distance, and is a special case of Minkowski distance for $p = 1$. The Minkowski distance f for the vectors o_i and o_j is defined as follows:

$$f(o_i, o_j) = (\sum_{a \in A} |o_i(a) - o_j(a)|^p)^{1/p}$$

where $p \in \mathbb{R}$. It is possible to consider using a distance measure for $p > 1$. The literature[80] suggests that this may be beneficial for attributes that are continuous and represent similar attributes such as height, width, and depth. However, this is beyond the scope of this dissertation.

follows:

$$f_H(o_i, o_j) = \sum_{a \in A} \begin{cases} 0 & if \quad o_i(a) = o_j(a) \\ 1 & if \quad o_i(a) \neq o_j(a). \end{cases}$$

Property 6.1. For any two objects $o_i, o_j \in O$ evaluation of Hamming distance $f_H(o_i, o_j)$ requires $m = |A|$ comparison operations and *m-1* additions, giving an overall computational complexity of $O(m + m - 1) = O(m)$.

6.2.2 Mechanism of centroid candidates extraction

In this chapter, let the *extractCandidates* function determine the centroids of the multiset $\hat{E}_c^+(Exp_c)$ based on the Hamming distance, more precisely:

Definition 6.2. Given a macrostructure in a form of Hamming distance f_H, an output of $extractCandidates(\hat{E}_c^+(Exp_c))$ is defined as a set of such $o' \in O$ that fulfill a condition:

$$F_H(o', \hat{E}_c^+(Exp_c)) = \min_{o \in O} F_H(o, \hat{E}_c^+(Exp_c)) \tag{6.2.1}$$

that is they minimize the following sum:

$$F_H(o, \hat{E}_c^+(Exp_c)) = \sum_{e \in E_c^+} (k_e^+ \cdot f_H(o, e)). \tag{6.2.2}$$

The above condition is fulfilled by such objects o' for which a sum of f_H distance to other objects from E_c^+ (additionally weighted by multiplicity of respective objects in $\hat{E}_c^+(Exp_c)$) is lowest. In the centroid extraction task (as opposed to the medoid determination task, which is analyzed in later chapters), a representative is selected from all available space. This means that for selected objects it may happen that the multiplicity of $k_{o'}^+$ is 0, which is equivalent to the fact that for a given object o' there is no positive learning material about such an object and a given concept. The o' objects are therefore the centroids of the multiset $\hat{E}_c^+(Exp_c)$.

In order to find such objects o' one can naively follow a definition 6.2 in a direct way by performing an exhaustive search over a set O, and by calculating a sum (6.2.2) for every potential candidate $o \in O$.

Theorem 6.1. *Naive approach, that is, an exhaustive search over a set O in order to find objects satisfying a condition (6.2.1) formulated in definition 6.2 leads to an exponential computational complexity in relation to a number of attributes used for describing objects from O provided that a number of values of each attribute is non-trivial ($\geqslant 2$) and finite.*

Proof. Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c. To estimate the complexity from below, let us assume that all attributes $a \in A$ are binary (that is, they have exactly two distinct values each). Then a cardinality of the universe $|O| = 2^m$. A computational complexity of each calculation of $f_H(o,e)$ is $O(m)$ (see property 6.1). A function $f_H(o,e)$ needs to be evaluated for each pair of objects $o \in O$ and $e \in E_c^+$, which leads to $n \cdot 2^m$ distinct evaluations. It leads to a final polynomial-exponential computational complexity $O(m \cdot n \cdot 2^m)$ with an exponential component dependent on m.

Naturally, as a number of values of any attribute $a \in A$ is greater than 2, a cardinality of the universe O grows, which translates to a higher number of required evaluations of f_{WH} and, in consequence, to higher complexity (e.g., if every attribute has exactly three distinct values, the final complexity becomes $O(m \cdot n \cdot 3^m)$).

In general, a component 2^m is a multiplication of cardinalities of sets of values of respective properties. To estimate the complexity from above, let us assume that l_{max} is the largest cardinality of sets of properties' values (it exists since there is always a maximum integer in a finite set of finite integers). It leads to an approximation of the upper bound of the complexity as $O(m \cdot n \cdot (l_{max})^m)$, which is still exponential in relation to m. ■

The use of exponential complexity algorithms in practical context is very limiting and often imposes strict restrictions on the size of the problem. In particular, it would be necessary to provide input data instances containing only a small number of attributes.

Let us investigate the way that for each object o that is the potential candidate for prototype, the sum $F_H(o, \hat{E}_c^+(Exp_c))$ is calculated:

$$F_H(o, \hat{E}_c^+(Exp_c)) = \sum_{e \in E_c^+} (k_e^+ \cdot f_H(o,e)) = \sum_{e \in E_c^+} \left(k_e^+ \cdot \sum_{a \in A} \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right).$$

It is valid to put k_e^+ inside the internal sum. Furthermore, both sums are finite, so the order of these sums is not important, thus

$$F_H(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[\sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right) \right]. \tag{6.2.3}$$

It is easy to notice that for a given $\hat{E}_c^+(Exp_c)$ and attribute a, the value of the expression within square brackets, i.e.

$$\sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right), \tag{6.2.4}$$

depends only on a choice of $o(a)$ and on values of particular $e(a)$ of particular objects from the original multiset and, what is more important, it does not depend on values of other attributes, both within object o, and within other objects e. It means that when choosing from the whole universe O, that is, when all value combinations are allowed, we can optimize (6.2.4) for every property separately.

Let as assume that $v \in V_a$ is one of values of an attribute a and let us introduce a notion

$$S(v, \hat{E}_c^+(Exp_c), a) = \sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad v = e(a) \\ 1 & if \quad v \neq e(a) \end{cases} \right). \tag{6.2.5}$$

Lemma 6.2. *A necessary and sufficient condition for an object $o' = (o'(a_1), o'(a_2), \ldots, o'(a_m))$ to be a proper centroid (satisfying (6.2.1)) is to fulfill a set of sub-conditions:*

$$S(o'(a_i), \hat{E}_c^+(Exp_c), a_i) = \min_{v \in V_{a_i}} S(v, \hat{E}_c^+(Exp_c), a_i) \tag{6.2.6}$$

for $i = 1, 2, \ldots, m$.

Proof. It follows directly from an argumentation preceding the lemma. ■

Lemma 6.3. *In order for o' to minimize $S(v, \hat{E}_c^+(Exp_c), a)$ it is enough for every particular attribute $a \in A$ to assign the most common (in objects from the original multiset) value of an attribute a to $o'(a)$. If there is more than one value of a that is most common in $\hat{E}_c^+(Exp_c)$, then all these values are equally good values of $o'(a)$.*

Proof. The expression $S(v, \hat{E}_c^+(Exp_c), a)$ for a respective attribute a directly evaluates to a number of objects from the multiset $\hat{E}_c^+(Exp_c)$ that exhibit a different value of a than v. It means that this expression is minimized by such values $v \in V_a$ that are most frequent in objects from the multiset $\hat{E}_c^+(Exp_c)$. ■

A commonsense interpretation of 'good candidates' described in lemma 6.3 is a vector containing most frequent values for each separate attribute. Attributes with multiple most-frequent values (that tied in terms of their occurrences) lead to additional optimal solutions which in some cases might lead to huge solution sets, e.g., in an initial stage of an agent's life where categories are not yet well defined.

Nonetheless, lemma 6.3 directly leads to a proposal of a three-phase algorithm 6.1. In the first phase (lines 1-5) it calculates occurrences of particular values within respective attributes. In the second phase (lines 6-7) the best values are chosen for each attribute (depending on calculated occurrences in Hamming distance). In the third phase (lines 8-17) a set of solutions (here: candidates) is constructed.

As stated in theorem 6.1, a naive approach to a Hamming distance-based evaluation of a centroid of a multiset has a mixed polynomial-exponential complexity between $O(m \cdot n \cdot 2^m)$ and $O(m \cdot n \cdot (l_{max})^m)$. Let us show the reduced computational complexity

Algorithm 6.1: *extractCandidates* for Hamming distance f_H

Input: a multiset $\hat{E}_c^+(Exp_c)$, a distance function f_H
Output: set of candidates $Candidates$

```
1  foreach attribute a in A do
2      Initiate a dictionary L_a = {⟨v, 0⟩ : v ∈ V_a};
3  foreach object o in E_c^+ do
4      foreach attribute a in A do
5          L_a(o(a)) ← L_a(o(a)) + k_o^+;
6  foreach attribute a in A do
7      H_a ← a set of values v from V_a with the highest number of occurrences
         L_a(v);
8  o ← new object;
9  Candidates ← {o};
10 foreach attribute a in A do
11     newCandidates ← ∅;
12     foreach value h in H_a do
13         foreach object o in Candidates do
14             o' ← duplicate of o;
15             o'(a) ← h;
16             add o' to newCandidates;
17     Candidates ← newCandidates;
```

provided by algorithm 6.1. The algorithm improves on this by reducing the number of loops and changing their order in comparison to the naive count.

Lemma 6.4. *A computational complexity of the first stage (lines 1-5) of algorithm 6.1 is approximated from above by* $O(m \cdot l_{max}) + O(n \cdot m) = O(m \cdot (l_{max} + n))$.

Proof. Initialization of dictionary L_a (lines 1-2) has to be performed for each of m attributes. In worst-case scenario, the number of values for an attribute is assumed to be equal to l_{max}, thus computational complexity of lines 1-2 is $O(m \cdot l_{max})$.

Calculation of frequency of values in objects in E_c^+ (lines 3-5) involves two loops - iterating over n objects and m attributes. Thus computational complexity of lines 3-5 is $O(n \cdot m)$.

It lets us evaluate an overall complexity of lines 1-5 as $O(m \cdot l_{max}) + O(n \cdot m) = O(m \cdot (l_{max} + n))$. ■

Lemma 6.5. *A computational complexity of the second stage (lines 6-7) of algorithm 6.1 is* $O(m \cdot l_{max})$.

Proof. A loop in line 6 iterates m times. The choice of the most frequent values is approximated as $O(l_{max})$ – their further assignment to H_a does not further raise a character of this complexity. It lets us evaluate an overall computational complexity of lines 6-7 as $O(m \cdot l_{max})$. ■

A complexity of the third stage (lines 8-17) of algorithm 6.1 requires a more extended comment. It is easy to notice that a set of optimal solutions (following the notions used in the algorithm) takes a form of a following Cartesian product:

$$Candidates = H_{a_1} \times H_{a_2} \times \cdots \times H_{a_m} \tag{6.2.7}$$

and its size is directly dependent on a cardinality of particular sets H_a. It means that in a borderline case where there is exactly one most common value per attribute, there is exactly one optimal solution. However, as a number of 'ties' grows within particular attributes, a set *Candidates* grows exponentially in size (per each attribute with more than one optimal value). It leads to an exponential evaluation of a worst-case complexity for the third stage of the algorithm (note that in our assumptions all optimal solutions need to be listed and returned in their final form; were it instead required to provide just one optimal solution, it would become as simple as $O(m)$).

Lemma 6.6. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c. As elements of sets H_a are chosen from corresponding sets V_A, then their cardinality cannot be larger than their respective counterparts. In consequence, $|H_a| \leqslant |V_a|$ for $a \in A$, and therefore it is possible to estimate a cardinality of all H_a from above by $l_{max} = \max_{a \in A} |V_a|$. A computational complexity of a third stage (lines 8-17) of algorithm 6.1 is approximated from above by $O(l_{max}^m)$ for the worst-case scenario.*

Proof. The complexity follows directly from a need to generate a whole Cartesian product (6.2.7) and an estimation mentioned in the lemma itself. It leads to the following dependencies:

$$\prod_{a \in A} |H_a| \leqslant \prod_{a \in A} |V_a| \leqslant \prod_{a \in A} \left(\max_{a' \in A} |V_{a'}| \right) = \prod_{a \in A} l_{max} = l_{max}^m.$$

■

A computational complexity of the worst-case scenario, as already mentioned, is extremely higher in comparison to the complexity of the best-case scenario which is summed up within a following straightforward lemma.

Lemma 6.7. *Assume $m = |A|$ – a number of attributes. If every H_a contains just a single element, that is if there is a single most frequent value for each attribute $a \in A$, then a computational complexity of a third stage (lines 8-17) of algorithm 6.1 is polynomial, $O(m)$.*

Proof. If there is a single most frequent value for each attribute $a \in A$, then $Candidates$ set includes only one element through all iterations of algorithm. The construction of that element needs m assignment operations. Thus computational complexity of lines 8-17 is $O(m)$. ■

It leads to the final estimation of a computational complexity of the whole algorithm 6.1.

Theorem 6.8. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c, $l_{max} = \max_{a \in A} |V_a|$ – an overestimation of the cardinality of sets V_a for $a \in A$.*

A computational complexity of the whole algorithm in the best-case scenario (with only one optimal candidate) is not worse than $O(m \cdot (l_{max} + n) + m \cdot l_{max} + m) = O(m \cdot (l_{max} + n))$.

A computational complexity of the whole algorithm in the worst-case scenario (with all potential candidates being optimal) is not worse than $O(m \cdot (l_{max} + n) + m \cdot l_{max} + l_{max}^m) = O(m \cdot (l_{max} + n) + l_{max}^m)$.

Proof. A proof follows directly from previous evaluations of computational complexities of particular parts which are simply sequentially connected in algorithm 6.1. ■

Conclusions. Two extreme situations regarding the computational complexity of the algorithm 6.1 are presented above. In practical applications, it is unlikely that many attributes have more than one most common value, so the computational complexity of the algorithm should approach the polynomial. The algorithm 6.1 can therefore be used in practice, which was confirmed by the simulation tests presented in chapter 7.

Worth noting is the fact, that compared to the naive approach, characterized by absolutely exponential computational complexity, the obtained algorithm reviews the original learning material (the multiset $\hat{E}_c^+(Exp_c)$) only once, and a potential exponential complexity element occurs only in the pessimistic case, when the size of the set of optimal solutions is very numerous and is related only to the construction of all instances of optimal candidates, not to their determination.

In a particular version of the algorithm 4.2, where only one optimal candidate would be searched and not the whole set of optimal candidates, the computational complexity of the proposed computational approach would be strictly polynomial.

6.2.3 Illustrative example – centroids

Let us assume, for the sake of simplicity, that the agent's experience consists of only two episodes. The world accessible to the agent's perception is described by five binary attributes. Its observations collected during two episodes are presented in the table below. As can be seen, they concern only one category c.

episode	**object**	a_1	a_2	a_3	a_4	a_5	**labels**
$Episode(t_1)$	$x_{1,1}$	1	1	1	1	1	*is-c*
	$x_{1,2}$	0	0	1	1	0	*is-c*
	$x_{1,3}$	0	0	1	0	1	*not-c*
	$x_{1,4}$	0	1	0	1	1	*not-c*
$Episode(t_2)$	$x_{2,1}$	1	1	0	1	1	*is-c*
	$x_{2,2}$	0	1	1	0	0	*is-c*
	$x_{2,3}$	0	0	1	1	1	*not-c*
	$x_{2,4}$	0	1	1	0	1	*not-c*

Table 6.1: Example using Hamming distance.

Initial computations Based on the above observations, as a result of the preprocessing performed by the agent after the second episode, its experience regarding the category c will be collected in the form of a multiset $Exp_c = \{1(o_7, +)^c, 1(o_{13}, +)^c, 1(o_{28}, +)^c, 1(o_{32}, +)^c, 1(o_6, -)^c, 1(o_8, -)^c, 1(o_{12}, -)^c, 1(o_{14}, -)^c\}$ where $o_6 = [00101]$, $o_7 = [00110]$, $o_8 = [00111]$, $o_{12} = [01011]$, $o_{13} = [01100]$, $o_{14} = [01101]$, $o_{28} = [11011]$, $o_{32} = [11111]$. Let us notice, that $O = \{o_1, ..., o_{32}\}$.

It follows from the above:

- $E^+ = E_c^+(Exp_c) = \{o_7, o_{13}, o_{28}, o_{32}\}$,
- $E^- = E_c^-(Exp_c) = \{o_6, o_8, o_{12}, o_{14}\}$.

Naive extraction of candidates In order to find candidates by naively following the definition 6.2, one has to perform an exhaustive search over the set O, involving calculation of the sum (6.2.2) for every potential candidate $o \in O$. Most important data produced during that process are shown in the table 6.2, naturally complete table consists of 32 rows.

$o \in O$	$f_H(o, o_7)$	$f_H(o, o_{13})$	$f_H(o, o_{28})$	$f_H(o, o_{32})$	$F_H(o, \hat{E}_c^+(Exp_c))$
o_1	2	2	4	5	13
o_2	3	3	3	4	13
			...		
o_{15}	1	1	3	2	7
o_{16}	2	2	2	1	7
			...		
o_{31}	2	2	2	1	7
o_{32}	3	3	1	0	7

Table 6.2: Exhaustive search for centroids, Hamming distance.

In this example $m = 5$, $n = 4$ and $l_{max} = 2$. From previous analysis, we know that it takes $(2 \cdot m - 1) \cdot n \cdot (l_{max})^m + (n-1) \cdot (l_{max})^m = (2 \cdot 5 - 1) \cdot 4 \cdot 2^5 + (4-1) \cdot 2^5 = 1248$ basic (like comparison and addition) operations to perform an exhaustive search over the set O.

Minimal $F_H(o, \hat{E}_c^+(Exp_c))$ is equal to 7. There are four objects $o' \in O$ with that value of $F_H(o, \hat{E}_c^+(Exp_c))$. They constitute a set of candidates for prototype: $Candidates = \{o_{15}, o_{16}, o_{31}, o_{32}\}$, where $o_{15} = [01110]$, $o_{16} = [01111]$, $o_{31} = [11110]$, $o_{32} = [11111]$.

Optimized extraction of candidates During the first stage of the algorithm 6.1 the number of occurrences of values of attributes is calculated, and stored in the dictionary L_a. The result is shown in the table 6.3. It takes $m \cdot l_{max} = 5 \cdot 2 = 10$ assignment operations to initialize dictionary L_a and $n \cdot m = 4 \cdot 5 = 20$ addition operations to calculate the number of occurrences.

	a_1	a_2	a_3	a_4	a_5
$L_a(0)$	2	1	1	1	2
$L_a(1)$	2	3	3	3	2
H_a	{0, 1}	{1}	{1}	{1}	{0, 1}

Table 6.3: Dictionary L_a and most frequent values H_a, Hamming distance.

In the second stage of the algorithm 6.1 a set of values v from V_a with the highest number of occurrences is determined for all $a \in A$. The result is also shown in the table 6.3. It takes 7 assignment operations to memorize all H_a sets.

During the third stage of the algorithm 6.1, the candidates for prototype are constructed. Since there are four potential candidates, their construction takes 20 assignment operations. The result is obviously identical to naive approach described above: $Candidates = \{o_{15}, o_{16}, o_{31}, o_{32}\}$, where $o_{15} = [01110]$, $o_{16} = [01111]$, $o_{31} = [11110]$, $o_{32} = [11111]$. Let us note however, that obtaining this result takes much less basic operations than in case of naive approach – 57 in total, comparing to 1248 in naive approach. Naturally, the difference will be greater as the number of attributes m grows.

Calculation of prototype and quality of model The quality of the model of category strongly depends on the order in which the parameters for the candidates are calculated. If vector $o_{31} = [11110]$ is selected first, then the model's coefficients look like this:

- $\tau_c^+ = 2$,
- $\tau_c^- = 3$,
- $Core_c(o_{31}) = \{o_7, o_{13}, o_{28}, o_{32}\}$,

- $Outer_c(o_{31}) = \{o_6, o_8, o_{12}, o_{14}\}$,
- $Boundary_c(o_{31}) = \varnothing$.

In this case, the results achieved meet the condition for category c to be learned because $|Core_c(o_{31})| = 4 \geqslant 0 = |Boundary_c(o_{31}) \cap E^+|$. In consequence, the object o_{31} is assigned as the prototype $o_c^\star$ of the properly established category c with τ_c^+ and τ_c^- as its radii. The category c can be integrated with the ontological knowledge base.

The above model can be shown as in the figure 6.1. Obviously, the category is well defined – the model divides above examples into two subsets: belonging and not belonging to the category, thus the border is empty.

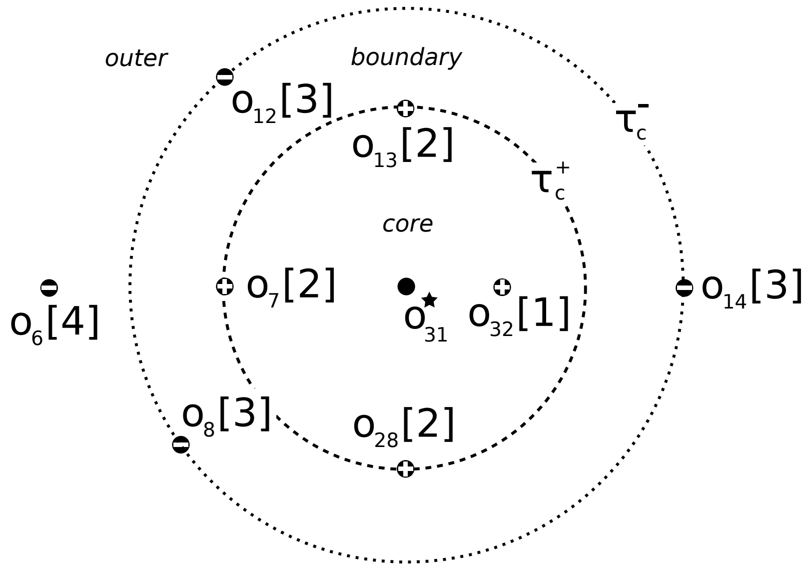


Figure 6.1: Model m_c for Hamming distance with centroid prototype.

However, if vector $o_{32} = [11111]$ is chosen as a candidate, the results will be much worse, the boundary includes as many as 6 objects:

- $\tau_c^+ = 1$,
- $\tau_c^- = NULL$,
- $Core_c(o_{32}) = \{o_{28}, o_{32}\}$,
- $Outer_c(o_{32}) = \varnothing$,
- $Boundary_c(o_{32}) = \{o_6, o_7, o_8, o_{12}, o_{13}, o_{14}\}$.

Nevertheless, the acceptance condition is met because $|Core_c(o_{32})| = 2 \geqslant 2 = |Boundary_c(o_{32}) \cap E^+|$. Consequently, the object will be considered a prototype, despite the apparently lower quality of the model. This version of the model is shown in the figure 6.2.

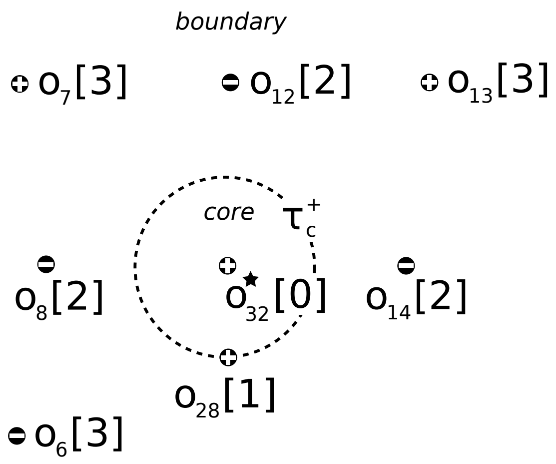


Figure 6.2: Model m_c for Hamming distance.

Grounding of statements The grounding of statements does not depend directly on the selection of the *extractCandidates* function, and it is quite similar regardless of the selection of the macrostructure. For the sake of completeness of example, a simple statement grounding for two objects will be shown, while more complicated cases are described in the chapter 5.

Let us assume, that in *Episode*(t_3) two objects are available to the agent's perception – their description is given in table 6.4. The objects are not marked with any labels by the teacher.

episode	**object**	a_1	a_2	a_3	a_4	a_5	**labels**
Episode(t_3)	$x_{3,1}$	0	0	0	1	1	
	$x_{3,2}$	1	1	1	0	0	

Table 6.4: Example for grounding using Hamming distance.

The object $x_{3,1}$ from working memory corresponds to object $o_4 = [00011]$ in embodied ontology and the object $x_{3,2}$ corresponds to object $o_{29} = [11100]$. Let us assume that the object $o_{31} = [11110]$ has been chosen as category's prototype $o_c^\star$.

Since $f_H(o_4, o_c^\star) = 4 \geqslant \tau_c^- = 3$, the object o_4 is located in the outer region of category's model and the epistemic satisfaction relation holds for statement $Know(x_{3,1} \notin c)$ (definition 5.6). Analogously, $f_H(o_{29}, o_c^\star) = 1 \leqslant \tau_c^+ = 2$, thus the object o_{29} is located in the core of category's model and the epistemic satisfaction relation holds for statement $Know(x_{3,2} \in c)$ (definition 5.5).

Conclusions. The method strongly depends on the order in which the candidates are selected. In future research, one could consider changing of the acceptance condition, or testing of all candidates, instead of accepting as a prototype the first one that meets the acceptance condition.

6.2.4 Mechanism of medoid candidates extraction

In this chapter, let the *extractCandidates* function determine the medoids of the multiset $\hat{E}_c^+(Exp_c)$ based on the Hamming distance, more precisely:

Definition 6.3. For a given multiset $\hat{E}_c^+(Exp_c)$ and a macrostructure in a form of Hamming distance function f_H, an output of the function $extractCandidates(\hat{E}_c^+(Exp_c))$ is defined as a set of such objects o' from E_c^+ that fulfill the condition:

$$F_H(o', \hat{E}_c^+(Exp_c)) = \min_{o \in E_c^+} F_H(o, \hat{E}_c^+(Exp_c)) \tag{6.2.8}$$

that is they minimize the following sum:

$$F_H(o, \hat{E}_c^+(Exp_c)) = \sum_{e \in E_c^+} (k_e^+ \cdot f_H(o, e)). \tag{6.2.9}$$

The definition 6.3 naturally resembles the definition 6.2, as it is still the definition of the central element of $\hat{E}_c^+(Exp_c)$ multiset (given as an input parameter) based on the Hamming distance. The difference is that the representative is selected from the multiset and not the entire object space O.

The above condition is met by objects $o' \in \hat{E}_c^+(Exp_c)$ for which the sum of Hamming distances to other objects in the multiset $\hat{E}_c^+(Exp_c)$ is the smallest. This translates into calculating the distance to the remaining[2] elements of the E_c^+ set, taking into account the frequency of occurrence of individual objects in the original $\hat{E}_c^+(Exp_c)$ set. The condition of selecting "from a set" is typical for medoids, and has a clear practical interpretation here, as it translates into the fact that potential representatives must be objects that have actually been observed by the agent and are present in a relevant part of the learning experience. This translates into the analytical condition $k_{o'}^+ > 0$.

Due to the fact, that only the multiset $\hat{E}_c^+(Exp_c)$ (effectively, only the set E_c^+ of unique objects of the given multiset) is searched for the central element, the problem

[2]Distances to other instances of identical objects may be omitted as they are equal to zero, from the definition of the metric.

of choosing a medoid assuming a naive approach should be less complex task than choosing a centroid.

To find objects o' we can refer directly to the definition 6.3, performing a full search of E_c^+, counting appropriate sum for each $o \in E_c^+$.

Theorem 6.9. *Let's assume that $\hat{E}_c^+(Exp_c)$ is the given multiset, $m = |A|$ is the cardinality of the set of attributes, and n denotes the cardinality of the set of positive examples for the category c, i.e. $n = |E_c^+|$. A full search of E_c^+ for medoids, i.e. objects that satisfy the condition from the definition 6.3 has the polynomial complexity of $O(m \cdot n^2)$.*

Proof. A computational complexity of each calculation of $f_H(o, e)$ is $O(m)$ (see property 6.1). Since the $f_H(o, e)$ function needs to be computed for each pair of $o, e \in E_c^+$ objects, it is necessary to perform $O(m \cdot n^2)$ basic operations. ■

Despite a polynomial computational complexity of a naive approach, it is worth showing that it can be further reduced. In order to proceed with the reduction, we need to analyze some properties related to medoids evaluated according to definition 6.3. While it is impossible to directly re-apply algorithm 6.1 previously used for the evaluation of centroids, an overall structure of the minimized distance functions remains the same. It is still a sum of Hamming distances between potential candidates and the remainder of the multiset. Therefore, let us start by repeating equation (6.2.3) which shows a transformed form of the sum which is being minimized, calculated for a potential candidate o.

$$F_H(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[\sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right) \right]. \tag{6.2.10}$$

Let us notice that for a given $\hat{E}_c^+(Exp_c)$ and assumed $a \in A$, the expression in the square brackets of (6.2.10) evaluates to a number of objects from the learning experience $\hat{E}_c^+(Exp_c)$ that exhibit a value **different** than v within the attribute a.

To reformulate this sum, let us introduce a following notion describing a number of objects from the learning experience $\hat{E}_c^+(Exp_c)$ that exhibit **the same** value v within the attribute a:

$$\widetilde{S}(v, \hat{E}_c^+(Exp_c), a) = \sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad v \neq e(a) \\ 1 & if \quad v = e(a) \end{cases} \right). \tag{6.2.11}$$

It leads us to a following re-formulation of the previous sum from (6.2.10):

$$\sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a). \end{cases} \right) = |\hat{E}_c^+(Exp_c)| - \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a).$$

Using this equality for a substitution in (6.2.10) we get the following:

$$F_H(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[|\hat{E}_c^+(Exp_c)| - \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right] =$$

$$= \sum_{a \in A} \left[|\hat{E}_c^+(Exp_c)| \right] - \sum_{a \in A} \left[\widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right],$$

where a component

$$\sum_{a \in A} \left[|\hat{E}_c^+(Exp_c)| \right] \tag{6.2.12}$$

is constant for given $\hat{E}_c^+(Exp_c)$ and assumed distance function (Hamming distance).

It entails that during the optimization (**minimization**) of $F_H(o, \hat{E}_c^+(Exp_c))$, the whole component (6.2.12) can be omitted, which leads to an equivalent minimization of the following expression:

$$- \sum_{a \in A} \left[\widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right],$$

which is, in turn, equivalent to the **maximization** of the following proposed criterion in its final form:

$$\widetilde{F}_H(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[\widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right]. \tag{6.2.13}$$

It further leads to the formulation of the following lemma:

Lemma 6.10. *For a given multiset $\hat{E}_c^+(Exp_c)$ and a macrostructure in a form of Hamming distance function f_H, objects o' satisfy definition 6.3 if and only if they satisfy the following condition*

$$\widetilde{F}_H(o', \hat{E}_c^+(Exp_c)) = \max_{o \in E_c^+} \widetilde{F}_H(o, \hat{E}_c^+(Exp_c)). \tag{6.2.14}$$

Proof. A proof consists of the whole already-presented deduction preceding the lemma. ■

This leads to the formulation of the practical three-phase algorithm 6.2. In the first phase (lines 1-5), for each attribute, the frequency of individual values of that attribute is calculated. In the second phase (lines 6-12), these frequencies are used to determine the value of the optimization criterion (at the same time, the maximum value of the criterion is determined – lines 11-12). The third phase (lines 13-16) constructs the target set of solutions.

Algorithm 6.2: *extractCandidates* – evaluation of medoids for distance f_H

Input: a multiset $\hat{E}_c^+(Exp_c)$, a distance function f_H
Output: set of candidates $Candidates$

1 **foreach** *attribute* a *in* A **do**
2   Initialize a dictionary $L_a = \{\langle v, 0\rangle : \ v \in V_a\}$;
3 **foreach** *object* o *in* E_c^+ **do**
4   **foreach** *attribute* a *in* A **do**
5         $L_a(o(a)) \leftarrow L_a(o(a)) + k_o^+$;
6 Initialize a 'maximal' value of a criterion as zero: $\widetilde{F}_{H,max} \leftarrow 0$;
7 **foreach** *object* o *in* E_c^+ **do**
8   Initialize a criterion for a current object: $\widetilde{F}_H(o, \hat{E}_c^+(Exp_c)) = 0$;
9   **foreach** *attribute* a *in* A **do**
10         $\widetilde{F}_H(o, \hat{E}_c^+(Exp_c)) \leftarrow \widetilde{F}_H(o, \hat{E}_c^+(Exp_c)) + L_a(o(a))$;
11   **if** $\widetilde{F}_H(o, \hat{E}_c^+(Exp_c)) > \widetilde{F}_{H,max}$ **then**
12         $\widetilde{F}_{H,max} \leftarrow \widetilde{F}_H(o, \hat{E}_c^+(Exp_c))$
13 Initialize $Candidates \leftarrow \varnothing$;
14 **foreach** *object* o *in* E_c^+ **do**
15   **if** $\widetilde{F}_H(o, \hat{E}_c^+(Exp_c)) == \widetilde{F}_{H,max}$ **then**
16   Add o to a set *Candidates*;

Remark 6.1. *The third stage (lines 13-16) of the algorithm could be incorporated into the second stage (lines 6-12), but this could lead to excessive juggling of a set Candidates, which would need to be purged as many times as the maximum value of the criterion is replaced in line 12.*

Let us proceed with an evaluation of a computational complexity of algorithm 6.2 in the presented form. The first stage of algorithm (lines 1-5) is identical to the first stage of algorithm 6.1, so its computational complexity is equal to $O(m \cdot (l_{max} + n))$ – for details refer to lemma 6.4 .

Lemma 6.11. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c, $l_{max} = \max_{a \in A} |V_a|$ – an overestimation of a cardinality of sets V_a for $a \in A$. A computational complexity of second stage of algorithm 6.2 (lines 6-12) is polynomial $O(n \cdot m)$.*

Proof. Line 6 has a complexity of $O(1)$. The loop in lines 7-12 is run n times. Inside that loop in line 8 we have variable initialization of complexity $O(1)$, in lines 9-10 there is an inner loop that executes m times (inside this loop, on line 10, a variable is incremented by the value of another variable, $O(1)$), and on lines 11-12 a conditional statement and an optional assignment operation (also $O(1)$). The computational complexity of the entire block (lines 6-12) can therefore be estimated at the level of $O(1 + n \cdot (1 + (m \cdot 1)) + 1 \cdot 1) = O(n \cdot m)$. ■

Lemma 6.12. *Let n denote the cardinality of the set of unique positive examples for the category c (that is $n = |E_c^+|$). The computational complexity of the third phase of the algorithm 6.2 (lines 13-16) is linear $O(n)$.*

Proof. Line 13 contains only initialization. In lines 14-16, the *for* loop is executed n times. Inside loop there is a comparison $O(1)$ and possibly addition of object to the set of solutions $O(1)$. So the complexity can be estimated at $O(1+n\cdot(1+1)) = O(n)$. ■

Theorem 6.13. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c, $l_{max} = \max_{a\in A} |V_a|$ – an overestimation of the cardinality of sets V_a for $a \in A$. A computational complexity of Algorithm 6.2 is polynomial $O(m \cdot (l_{max} + n))$.*

Proof. A proof follows directly from previous evaluations of computational complexities of particular parts which are simply sequentially connected in algorithm 6.2: $O(m \cdot (l_{max} + n)) + O(n \cdot m) + O(n) = O(m \cdot (l_{max} + n))$. ■

In practical applications $(l_{max} + n)$ component should be much lower than n^2 – e.g. in case where all attributes are binary $l_{max} = 2$, and $2 + n < n^2$ for all $n > 2$. Therefore computational complexity of algorithm 6.2 should be lower accordingly in comparison to the naive approach. This reduction of computational complexity from $O(m \cdot n^2)$ to $O(m \cdot (l_{max} + n))$ may have an influence on a fluent behavior of the system for large sets of agent's learning experience.

It is worth noticing, that in particular implementations many of the weighted sums naively computed in presented algorithms using *for* loops, could be implemented in a more effective way using hardware supporting fast matrix-based operations.

6.2.5 Illustrative example – medoids

Input data in this example are identical like in example using centroid candidates extraction method – they are shown in table 6.1. Therefore, in this chapter only key differences will be shown.

Initial computations Like in the previous example: $Exp_c = \{1(o_7, +)^c$, $1(o_{13}, +)^c$, $1(o_{28}, +)^c$, $1(o_{32}, +)^c$, $1(o_6, -)^c$, $1(o_8, -)^c$, $1(o_{12}, -)^c$, $1(o_{14}, -)^c\}$ where $o_6 = [00101]$, $o_7 =$

[00110], $o_8 = [00111]$, $o_{12} = [01011]$, $o_{13} = [01100]$, $o_{14} = [01101]$, $o_{28} = [11011]$, $o_{32} = [11111]$.

Also:

- $E^+ = E_c^+(Exp_c) = \{o_7, o_{13}, o_{28}, o_{32}\}$,
- $E^- = E_c^-(Exp_c) = \{o_6, o_8, o_{12}, o_{14}\}$.

Naive extraction of candidates In order to find candidates by naively following the definition 6.3, one has to perform an exhaustive search over the set E_c^+, involving calculation of the sum (6.2.9) for every potential candidate $o \in E_c^+$. Data produced during that process are shown in table 6.5.

$o \in O$	$f_H(o, o_7)$	$f_H(o, o_{13})$	$f_H(o, o_{28})$	$f_H(o, o_{32})$	$F_H(o, \hat{E}_c^+(Exp_c))$
o_7	0	2	4	3	9
o_{13}	2	0	4	3	9
o_{28}	4	4	0	1	9
o_{32}	3	3	1	0	7

Table 6.5: Exhaustive search for medoids, Hamming distance.

In this example $m = 5$ and $n = 4$. From previous analysis, we know that it takes $(2 \cdot m - 1) \cdot n^2 + (n - 1) \cdot n = (2 \cdot 5 - 1) \cdot 4^2 + (4 - 1) \cdot 4 = 156$ basic (like comparison and addition) operations to perform an exhaustive search over a set E_c^+. Let us note, that it is much lower number than in case of naive search for centroids, since in that example the set E_c^+ is much smaller than universe O.

Minimal $F_H(o, \hat{E}_c^+(Exp_c))$ is equal to 7. There is only one object $o' \in E_c^+$ with that value of $F_H(o, \hat{E}_c^+(Exp_c))$. It is the only member of the set of candidates for prototype: $Candidates = \{o_{32}\}$.

Optimized extraction of candidates During the first stage of algorithm 6.2 the number of occurrences of values of attributes is calculated, and stored in the dictionary L_a (just like in centroid case). The result is shown in table 6.6. It takes $m \cdot l_{max} = 5 \cdot 2 = 10$ assignment operations to initialize dictionary L_a and $n \cdot m = 4 \cdot 5 = 20$ addition operations to calculate the number of occurrences.

	a_1	a_2	a_3	a_4	a_5
$L_a(0)$	2	1	1	1	2
$L_a(1)$	2	3	3	3	2

Table 6.6: Dictionary L_a, Hamming distance.

In the second stage of algorithm 6.2 optimization criterion $\widetilde{F}_H(o, \hat{E}_c^+(Exp_c))$ is calculated for each object $o \in E_c^+$. The result is:

- $\widetilde{F}_H(o_7, \hat{E}_c^+(Exp_c)) = 11$
- $\widetilde{F}_H(o_{13}, \hat{E}_c^+(Exp_c)) = 11$
- $\widetilde{F}_H(o_{28}, \hat{E}_c^+(Exp_c)) = 11$
- $\widetilde{F}_H(o_{32}, \hat{E}_c^+(Exp_c)) = 13$.

from that $\widetilde{F}_{H,max} = 13$. It takes $n \cdot m = 4 \cdot 5 = 20$ addition operations to calculate optimization criteria.

During the third stage of algorithm 6.2 candidates for prototype are chosen. It takes $n = 4$ comparation operations to find them. The result is obviously identical to the naive approach described above: $Candidates = \{o_{32}\}$. Obtaining this result takes less basic operations that in the case of naive approach – 54 in total, comparing to 156 in the naive approach. Let us note however, that this example is very small, for the sake of simplicity. Naturally, there would be much bigger difference in favor of optimized approach if n is bigger, which should be the case in practical applications.

Calculation of prototype and quality of the model The results for candidate $o_{32} = [11111]$ where already shown for the centroid case:

- $\tau_c^+ = 1$,
- $\tau_c^- = NULL$,
- $Core_c(o_{32}) = \{o_{28}, o_{32}\}$,
- $Outer_c(o_{32}) = \varnothing$,
- $Boundary_c(o_{32}) = \{o_6, o_7, o_8, o_{12}, o_{13}, o_{14}\}$.

The acceptance condition is fulfilled, because $|Boundary_c(o_{32}) \cap E^+| = 2$ and $|Core_c(o_{32})| = 2$ thus $|Core_c(o_{32})| \geqslant |Boundary_c(o_{32}) \cap E^+|$. In consequence, o_{32} will be set as a prototype of category c. This model was shown in the figure 6.2.

Clearly, in this example the set of candidates is much smaller in medoids case, than in centroids case. As a result, the quality of model in medoids case may be worse than in centroids case, depending on which of the centroid candidates will be chosen. On the other hand, searching for medoid candidates is less computationally expensive, especially in the case of naive search.

6.3 Weighted Hamming distance

An analysis of relational datasets in a typical data mining workflow usually includes feature selection. It is widely accepted that some features may have higher impact on the phenomena in question or, in general, they seem to carry more information than the others. While dropping properties from objects' descriptions is not examined in this dissertation, we will investigate an assignment of importance to particular properties. Considering Hamming distance, this can be done in a numerical way by a simple addition of weights, as shown in the following definition of Weighted Hamming distance (WH).

The following chapter contains description of algorithms and ideas very similar to the ones from chapter about unweighted Hamming distance. It might seem repetitive, but adding weights to Hamming distance forces small, but very important modifications to the algorithms. Therefore they have to be analyzed separately. In addition, giving even the same considerations, makes it possible to understand the content of a chapter without going back to previous chapters.

6.3.1 Macrostructure

Definition 6.4. The Weighted Hamming distance f_{WH} for the vectors o_i and o_j is defined as follows:

$$f_{WH}(o_i, o_j) = \sum_{a \in A} \begin{cases} 0 & if \quad o_i(a) = o_j(a) \\ w_a & if \quad o_i(a) \neq o_j(a). \end{cases}$$

where $w_a \geqslant 0$ denotes the weight[3] of the attribute $a \in A$.

Property 6.2. For any two objects $o_i, o_j \in O$ evaluation of Weighted Hamming distance $f_{WH}(o_i, o_j)$ requires $m = |A|$ comparison operations, m assignments, and *m-1* additions, giving an overall computational complexity of $O(m + m + m - 1) = O(m)$.

In this model, the introduction of weights is based on the assumption that not all the attributes for a given category are equally important. For example, it seems obvious that for an object in the process of including it in the *bird* category, the fact of having a beak is more important than the black color of the feathers. Attribute weights should be based on empirical data. Basing on psycholinguistic research, it seems to be beneficial if they support the condition (b) given in the chapter 6.1.1, and thus allow for easier distinguishing of non-category objects from category objects.

For the above reasons, the information gain was assumed as the weight of the a attribute, which corresponds to the entropy reduction when dividing the set of examples into $|V_a|$ subsets according to the value of the a attribute, where $|V_a|$ is the number of values of attribute a. It is a measure often used for decision tree induction[80].

[3]In literature sometimes Weighted Hamming distance is normalized[39, 100].

Definition 6.5. Given a collection S, containing positive and negative examples of some target concept, the entropy of S relative to this Boolean classification is defined as[54]:

$$Entropy(S) = -p_{\oplus}\, log_2\, p_{\oplus} - p_{\ominus}\, log_2\, p_{\ominus}$$

where $p_{\oplus}$ is the proportion of positive examples in S and $p_{\ominus}$ is the proportion of negative examples in S. In all calculations involving entropy we define $0\, log_2\, 0$ to be 0.

Definition 6.6. Information gain $Gain(S, a)$ of an attribute a relative to collection of objects S, is defined as[54]:

$$Gain(S, a) = Entropy(S) - \sum_{v \in V_a} \frac{|S_v|}{|S|} Entropy(S_v)$$

where V_a is the set of all possible values for attribute a, and S_v is the subset of S for which attribute a has value v (i.e. $S_v = \{s \in S : s(a) = v\}$). Note the first term in equation is just the entropy of the original collection S, and the second term is the expected value of the entropy after S is partitioned using attribute a.

In the following chapters, it is assumed that the weight of attribute a for the model of category c is equal to the information gain for that attribute, calculated for the multiset Exp_c representing the agent's experience concerning category c:

$$w_a = Gain(Exp_c, a).$$

Example 4. Let us assume a microstructure of all objects o from a cognitive universe O in a form of fixed-length binary vectors, where length $n = 2$. $O = \{o_1, o_2, o_3, o_4\}$ where $o_1 = [00]$, $o_2 = [01]$, $o_3 = [10]$, $o_4 = [11]$. Set of attributes is denoted as $A = (a_1, a_2)$.

Let us assume that $Exp_c = \{2(o_1, +)^c, 4(o_2, +)^c, 2(o_3, +)^c, 0(o_4, +)^c, 0(o_1, -)^c,$ $0(o_2, -)^c, 4(o_3, -)^c, 4(o_4, -)^c\}$. In total, there are 16 objects in multiset Exp_c: 8 of them are positive and 8 of them are negative. Thus:

$$Entropy(Exp_c) = -\frac{8}{16}\, log_2 \frac{8}{16} - \frac{8}{16}\, log_2 \frac{8}{16} = 1$$

Let us denote $S_{1,0} = \{(o, +)^c, (o, -)^c \in Exp_c : o(a_1) = 0\}$ and $S_{1,1} = \{(o, +)^c, (o, -)^c \in Exp_c : o(a_1) = 1\}$. Entropy for these sets is calculated as follows:

$$Entropy(S_{1,0}) = -\frac{6}{6}\, log_2 \frac{6}{6} - \frac{0}{6}\, log_2 \frac{0}{6} = 0$$

$$Entropy(S_{1,1}) = -\frac{2}{10}\, log_2 \frac{2}{10} - \frac{8}{10}\, log_2 \frac{8}{10} \approx 0.722$$

and finally:

$$Gain(Exp_c, a_1) = Entropy(Exp_c) - \frac{|S_{1,0}|}{|Exp_c|} Entropy(S_{1,0}) - \frac{|S_{1,1}|}{|Exp_c|} Entropy(S_{1,1})$$

$$= 1 - \frac{6}{16} \cdot 0 - \frac{10}{16} \cdot 0.722 \approx 0.549$$

Let us denote $S_{2,0} = \{(o,+)^c, (o,-)^c \in Exp_c : o(a_2) = 0\}$ and $S_{2,1} = \{(o,+)^c, (o,-)^c \in Exp_c : o(a_2) = 1\}$. Entropy for these sets is calculated as follows:

$$Entropy(S_{2,0}) = -\frac{4}{8} log_2 \frac{4}{8} - \frac{4}{8} log_2 \frac{4}{8} = 1$$

$$Entropy(S_{2,1}) = -\frac{4}{8} log_2 \frac{4}{8} - \frac{4}{8} log_2 \frac{4}{8} = 1$$

and finally:

$$Gain(Exp_c, a_2) = Entropy(Exp_c) - \frac{|S_{2,0}|}{|Exp_c|} Entropy(S_{2,0}) - \frac{|S_{2,1}|}{|Exp_c|} Entropy(S_{2,1})$$

$$= 1 - \frac{8}{16} \cdot 1 - \frac{8}{16} \cdot 1 = 0$$

To sum up, weight of attribute a_1 is equal to $Gain(Exp_c, a_1) \approx 0.549$ and weight of attribute a_2 is equal to $Gain(Exp_c, a_2) = 0$. Weight of a_1 is much bigger than weight of a_2, because a_1 is much better than a_2 in separating of positive from negative objects.

Property 6.3. Assume $m = |A|$ – a number of attributes, $n = |Exp_c|$ – a cardinality of the set of objects for model of category c, l_{max} – the largest cardinality of sets of attributes' values. Assuming that the weight of attribute a for the model of category c is equal to the information gain for that attribute, a computational complexity of calculation of a set of weights for all attributes in A is equal to $O(m \cdot (n + l_{max}))$.

6.3.2 Mechanism of centroid candidates extraction

In this chapter, let the *extractCandidates* function determine the centroids of the multiset $\hat{E}_c^+(Exp_c)$ based on the Weighted Hamming distance, more precisely:

Definition 6.7. Given a macrostructure in a form of Weighted Hamming distance f_{WH}, an output of $extractCandidates(\hat{E}_c^+(Exp_c))$ is defined as a set of such $o' \in O$ that fulfill a condition:

$$F_{WH}(o', \hat{E}_c^+(Exp_c)) = \min_{o \in O} F_{WH}(o, \hat{E}_c^+(Exp_c)) \tag{6.3.1}$$

that is they minimize the following sum:

$$F_{WH}(o, \hat{E}_c^+(Exp_c)) = \sum_{e \in E_c^+} (k_e^+ \cdot f_{WH}(o, e)). \tag{6.3.2}$$

The above condition is fulfilled by such objects o' for which a sum of Weighted Hamming distance to other objects from E_c^+ (additionally weighted by multiplicity of respective objects in $\hat{E}_c^+(Exp_c)$) is the lowest. In the centroid extraction task (as opposed to the medoid determination task, which is analyzed in later chapters), a representative is selected from all available space. This means that for selected objects it may happen that the multiplicity of $k_{o'}^+$ is 0, which is equivalent to the fact that for a given o' object there is no positive learning material about such an object and a given concept. The o' objects are therefore the centroids of the multiset $\hat{E}_c^+(Exp_c)$. In order to find such objects o' one can naively follow the definition 6.7 in the direct way by performing an exhaustive search over the set O, and by calculating the sum (6.3.2) for every potential candidate $o \in O$.

Theorem 6.14. *Naive approach, that is, an exhaustive search over the set O in order to find objects satisfying the condition (6.3.1) formulated in definition 6.7 leads to an exponential computational complexity in relation to a number of attributes used for describing objects from O provided that a number of values of each attribute is non-trivial ($\geqslant 2$) and finite.*

Proof. Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c. To estimate the complexity from below, let us assume that all attributes $a \in A$ are binary (that is, they have exactly two distinct values each). Then the cardinality of the universe $|O| = 2^m$. A computational complexity of each calculation of $f_{WH}(o, e)$ is $O(m)$ (see property 6.2). A function $f_{WH}(o, e)$ needs to be evaluated for each pair of objects $o \in O$ and $e \in E_c^+$, which leads to $n \cdot 2^m$ distinct evaluations. It leads to a final polynomial-exponential computational complexity $O(m \cdot n \cdot 2^m)$ with an exponential component dependent on m.

Naturally, as a number of values of any attribute $a \in A$ is greater than 2, the cardinality of the universe O grows, which translates to a higher number of required evaluations of f_{WH} and, in consequence, to higher complexity (e.g., if every attribute as exactly three distinct values, the final complexity becomes $O(m \cdot n \cdot 3^m)$).

In general, a component 2^m is a multiplication of cardinalities of sets of values of respective properties. To estimate the complexity from above, let us assume that l_{max} is the largest cardinality of sets of properties' values (it exists since there is always a maximum integer in a finite set of finite integers). It leads to an approximation of the upper bound of the complexity as $O(m \cdot n \cdot (l_{max})^m)$, which is still exponential in relation to m. ∎

The use of exponential complexity algorithms in a practical context is very limiting and often imposes strict restrictions on the size of the problem. In particular, it would be necessary to provide input data instances containing only a small number of attributes.

Learning from experience related to the Hamming distance, let us check a structure of f_{WH} evaluation for a candidate o and a multiset $\hat{E}_c^+(Exp_c)$):

$$F_{WH}(o, \hat{E}_c^+(Exp_c)) = \sum_{e \in E_c^+} (k_e^+ \cdot f_{WH}(o, e)) = \sum_{e \in E_c^+} \left(k_e^+ \cdot \sum_{a \in A} \begin{cases} 0 & if \quad o(a) = e(a) \\ w_a & if \quad o(a) \neq e(a) \end{cases} \right).$$

Similar transformations to the ones performed in (6.2.3) lead to the following equality:

$$F_{WH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[\sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ w_a & if \quad o(a) \neq e(a) \end{cases} \right) \right]. \tag{6.3.3}$$

Since w_a is constant for a given a, not only can we put it in front of the curly bracket

$$\begin{cases} 0 & if \quad o(a) = e(a) \\ w_a & if \quad o(a) \neq e(a) \end{cases} = w_a \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases}$$

but we can also extract it from the interior of the corresponding sub-sum, leading to the following equality:

$$\sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ w_a & if \quad o(a) \neq e(a) \end{cases} \right) = w_a \cdot \sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right).$$

It leads to the following reformulation of (6.3.3):

$$F_{WH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[w_a \cdot \sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right) \right]. \tag{6.3.4}$$

It is easy to notice that for a given $\hat{E}_c^+(Exp_c)$, for an assumed a, and provided that values of weights w_a are finite and greater than *0*, a value of the expression within square brackets of (6.3.4), that is,

$$w_a \cdot \sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right), \tag{6.3.5}$$

depends only on the choice of $o(a)$ and on values of particular $e(a)$ of particular objects from the original multiset and, what is more important, it does not depend on values of other attributes, both within object o, and within other objects e. In a borderline case for the particular $a \in A$ the weight w_a might be equal to zero. In such a situation the value of respective $o(a)$ does not have any influence on (6.3.4) as the whole (6.3.5) is reduced to 0 for that a.

In terms of interpretation, an expression (6.3.5) aggregates an overall influence related to the attribute a on the final value of f_{WH}. It means that when choosing from the whole universe O, that is, when all combinations of values are allowed, we can optimize (6.3.5) for every property separately. Each particular w_a is constant. The borderline case of $w_a = 0$ has been already described. If w_a is, however, greater than 0, it can be omitted during optimization. Let us remind the notion introduced in (6.2.5), given that $v \in V_a$ is one of values of the attribute a:

$$S(v, \hat{E}_c^+(Exp_c), a) = \sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad v = e(a) \\ 1 & if \quad v \neq e(a) \end{cases} \right).$$

which makes the following lemma convenient to formulate:

Lemma 6.15. *A necessary and sufficient condition for the object $o' = (o'(a_1), o'(a_2), \ldots, o'(a_m))$ to be the proper centroid (satisfying (6.3.1)) is to fulfill the set of sub-conditions:*

$$\begin{cases} o'(a_i) \in V_{a_i} \quad \textit{iff} \quad w_{a_i} = 0, \\ S\Big(o'(a_i), \hat{E}_c^+(Exp_c), a_i\Big) = \min_{v \in V_{a_i}} S\Big(v, \hat{E}_c^+(Exp_c), a_i\Big) \quad \textit{otherwise} \end{cases} \tag{6.3.6}$$

for $i = 1, 2, \ldots, m$.

Proof. It follows directly from the argumentation preceding the lemma. ■

Lemma 6.16. *In order for o' to minimize $S(w, \hat{E}_c^+(Exp_c), a)$ it is enough to do the following for every particular attribute $a \in A$:*

- *if the respective weight w_a is equal to zero, assign any value from V_a to $o'(a)$,*
- *otherwise, assign one of the most common values of the attribute a (in objects from the original multiset) to $o'(a)$.*

Proof. The case of $w_a = 0$ is trivial. If, however, $w_a > 0$ then notice that the expression (6.2.5) for the respective property a directly evaluates to a number of objects from a multiset $\hat{E}_c^+(Exp_c)$ that exhibit a different value of a than v. It means that the (6.2.5) is maximized by such values $v \in V_a$ that are most frequent in objects from a multiset $\hat{E}_c^+(Exp_c)$. ■

A common-sense interpretation of 'good candidates' described in lemma 6.16 is a vector containing most frequent values for each separate attribute assigned with a positive weight in the Weighted Hamming distance **or** with any of the allowed values of attributes assigned with $w_a = 0$. Multiple zero-weights in the Weighted Hamming distance and/or attributes with multiple most-frequent values (that tied in terms of their occurrences) lead to additional optimal solutions which in some cases might lead to huge solution sets, e.g., in an initial stage of an agent's life where categories are not yet well defined.

Nonetheless, lemma 6.16 directly leads to the proposal of the three-phase algorithm 6.3. In the first phase (lines 1-6) it calculates occurrences of particular values within respective attributes. In the second phase (lines 7-11) the best values are chosen for each attribute (depending on calculated occurrences and/or values of corresponding weights in Weighted Hamming distance). In the third phase (lines 12-21) a set of solutions (here: candidates) is constructed.

As stated in theorem 6.14, a naive approach to a Weighted Hamming distance-based evaluation of a centroid of a multiset has a mixed polynomial-exponential complexity between $O(m \cdot n \cdot 2^m)$ and $O(m \cdot n \cdot (l_{max})^m)$. Let us show the reduced computational complexity provided by algorithm 6.3.

A computational complexity of the first stage (lines 1-6) of algorithm 6.3 is relatively similar to a computational complexity of lines 1-5 of algorithm 6.1 except an additional *if*-statement is added to actually skip some evaluation-cycles of the loop for attributes where the weight $w_a = 0$.

Theorem 6.17. *A computational complexity of the first stage (lines 1-6) of algorithm 6.3 is approximated from above by* $O(m \cdot l_{max} + n \cdot m) = O(m \cdot (l_{max} + n))$.

Proof. Follows directly from previous proofs. ■

Theorem 6.18. *A computational complexity of the second stage (lines 7-11) of algorithm 6.3 is* $O(m \cdot l_{max})$.

Proof. A loop in line 7 cycles m times. The *if*-statement inside is $O(1)$ and it leads either to the choice of the most frequent values approximated itself as $O(l_{max})$ (their further assignment to H_a does not further raise a character of this complexity) or to an assignment of the whole V_a to H_a which is also not worse in comparison than $O(l_{max})$.

It lets us evaluate an overall complexity of lines 7-11 as $O(m \cdot 1 \cdot l_{max}) = O(m \cdot l_{max})$. ■

A complexity of the third stage (lines 12-21) of algorithm 6.3 requires a more extended comment. It is easy to notice that a set of optimal solutions (following the

Algorithm 6.3: *extractCandidates* – evaluation of centroids for Weighted Hamming distance f_{WH}

Input: a multiset $\hat{E}_c^+(Exp_c)$, a distance function f_{WH}, a set of weights w_a for all $a \in A$
Output: set of candidates $Candidates$

foreach *attribute* a *in* A **do**
  Initiate a dictionary $L_a = \{\langle v, 0\rangle : \ v \in V_a\}$;
foreach *object* o *in* E_c^+ **do**
  foreach *attribute* a *in* A **do**
    if $w_a > 0$ **then**
      $L_a(o(a)) \leftarrow L_a(o(a)) + k_o^+$;
foreach *attribute* a *in* A **do**
  if $w_a > 0$ **then**
    $H_a \leftarrow$ a set of values v from V_a with the highest number of occurrences;
  else
    $H_a \leftarrow V_a$;
$o \leftarrow$ new object;
$Candidates \leftarrow \{o\}$;
foreach *attribute* a *in* A **do**
  $newCandidates \leftarrow \varnothing$;
  foreach *value* h *in* H_a **do**
    foreach *object* o *in* $Candidates$ **do**
      $o' \leftarrow$ duplicate of o;
      $o'(a) \leftarrow h$;
      add o' to $newCandidates$;
  $Candidates \leftarrow newCandidates$;

notions used in the algorithm) takes a form of the following Cartesian product:

$$Candidates = H_{a_1} \times H_{a_2} \times \cdots \times H_{a_m} \tag{6.3.7}$$

and its size is directly dependent on the cardinality of particular sets H_a. It means that in a borderline case where there is exactly one most common value per attribute, there is exactly one optimal solution. However, as a number of 'ties' grows within particular attributes, a set *Candidates* grows exponentially in size (per each attribute with more than one optimal value). It leads to an exponential evaluation of the worst-case complexity for the third stage of the algorithm (note that in our assumptions all optimal solutions need to be listed and returned in their final form; were it instead required to provide just one optimal solution, it would become as simple as $O(m)$).

Lemma 6.19. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – the cardinality of the set of positive examples for the category c. As elements of sets H_a are chosen from corresponding sets V_A, then their cardinality cannot be larger than their respective counterparts. In consequence, $|H_a| \leqslant |V_a|$ for $a \in A$, and therefore it is possible to estimate the cardinality of all H_a from above by $l_{max} = \max_{a \in A} |V_a|$. A computational complexity of the third stage (lines 12-21) of algorithm 6.3 is approximated from above by $O(l_{max}^m)$ for the worst-case scenario.*

Proof. The complexity follows directly from the need to generate the whole Cartesian product (6.3.7) and an estimation mentioned in the lemma itself. It leads to the following dependencies:

$$\prod_{a \in A} |H_a| \leqslant \prod_{a \in A} |V_a| \leqslant \prod_{a \in A} \left(\max_{a' \in A} |V_{a'}| \right) = \prod_{a \in A} l_{max} = l_{max}^m.$$

■

A computational complexity of the worst case scenario, as already mentioned, is extremely higher in comparison to the complexity of the best case scenario which is summed up within a following straightforward lemma.

Lemma 6.20. *Assume $m = |A|$ – a number of attributes. If every H_a contains just a single element, that is if there is a single most frequent value for each attribute $a \in A$, then a computational complexity of the third stage (lines 12-21) of algorithm 6.3 is polynomial, $O(m)$.*

Proof. Straightforward. ■

It leads to the final estimation of the computational complexity of the whole algorithm 6.3.

Theorem 6.21. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c, $l_{max} = \max_{a \in A} |V_a|$ – an overestimation of the cardinality of sets V_a for $a \in A$.*

A computational complexity of the whole algorithm in the best case scenario (with only one optimal candidate) is not worse than $O(m \cdot (l_{max} + n) + m \cdot l_{max} + m) = O(m \cdot (l_{max} + n))$.

A computational complexity of the whole algorithm in the worst case scenario (with all candidates being optimal) is not worse than $O(m \cdot (l_{max} + n) + m \cdot l_{max} + l_{max}^m) = O(m \cdot (l_{max} + n) + l_{max}^m)$.

Proof. A proof follows directly from previous evaluations of computational complexities of particular parts which are simply sequentially connected in algorithm 6.3. ■

Conclusions. Two extreme situations regarding the computational complexity of the algorithm 6.3 are presented above. In practical applications, it is unlikely that many attributes have more than one most common value, so the computational complexity of the algorithm should approach the polynomial. The algorithm 6.3 can therefore be used in practice, which was confirmed by the simulation tests presented in chapter 7.

Worth noting is the fact, that compared to the naive approach, characterized by absolutely exponential computational complexity, the obtained algorithm reviews the original learning material (the multiset $\hat{E}_c^+(Exp_c)$) only once, and a potential exponential complexity element occurs only in the pessimistic case, when the size of the set of optimal solutions is very numerous and is related only to the construction of all instances of optimal candidates, not to their determination.

In the particular version of the algorithm 4.2, where only one optimal candidate would be requested and not the whole set of optimal candidates, the computational complexity of the proposed computational approach would be strictly polynomial.

6.3.3 Illustrative example – centroids

We assume that the agent's experience is identical to described in the chapter 6.2.3.

episode	**object**	a_1	a_2	a_3	a_4	a_5	**labels**
$Episode(t_1)$	$x_{1,1}$	1	1	1	1	1	*is-c*
	$x_{1,2}$	0	0	1	1	0	*is-c*
	$x_{1,3}$	0	0	1	0	1	*not-c*
	$x_{1,4}$	0	1	0	1	1	*not-c*
$Episode(t_2)$	$x_{2,1}$	1	1	0	1	1	*is-c*
	$x_{2,2}$	0	1	1	0	0	*is-c*
	$x_{2,3}$	0	0	1	1	1	*not-c*
	$x_{2,4}$	0	1	1	0	1	*not-c*

Table 6.7: Example using Weighted Hamming distance.

Initial computations Based on the above observations, as a result of the preprocessing performed by the agent after the second episode, its experience regarding the category c will be collected in the form of a multiset $Exp_c = \{1(o_7,+)^c, 1(o_{13},+)^c, 1(o_{28},+)^c, 1(o_{32},+)^c, 1(o_6,-)^c, 1(o_8,-)^c, 1(o_{12},-)^c, 1(o_{14},-)^c\}$ where $o_6 = [00101]$, $o_7 = [00110]$, $o_8 = [00111]$, $o_{12} = [01011]$, $o_{13} = [01100]$, $o_{14} = [01101]$, $o_{28} = [11011]$, $o_{32} = [11111]$. Let us note, that $O = \{o_1, ..., o_{32}\}$.

It follows from the above:

- $E^+ = E_c^+(Exp_c) = \{o_7, o_{13}, o_{28}, o_{32}\}$,
- $E^- = E_c^-(Exp_c) = \{o_6, o_8, o_{12}, o_{14}\}$.

Naive extraction of candidates For this example, the information gain (i.e. the weight of a given attribute) for the following attributes is calculated as follows[4]:

- $w_{a_1} = Gain(Exp_c, a_1) = 0.311$,
- $w_{a_2} = Gain(Exp_c, a_2) = 0.049$,
- $w_{a_3} = Gain(Exp_c, a_3) = 0$,
- $w_{a_4} = Gain(Exp_c, a_4) = 0.049$
- $w_{a_5} = Gain(Exp_c, a_5) = 0.311$.

In order to find candidates by naively following the definition 6.7, one has to perform an exhaustive search over the set O, involving calculation of the sum (6.3.2) for every potential candidate $o \in O$. Most important data produced during that process are shown in table 6.8, naturally complete table consists of 32 rows.

$o \in O$	$f_{WH}(o, o_7)$	$f_{WH}(o, o_{13})$	$f_{WH}(o, o_{28})$	$f_{WH}(o, o_{32})$	$F_{WH}(o, \hat{E}_c^+(Exp_c))$
o_1	0.049	0.049	0.720	0.720	1.538
o_2	0.360	0.360	0.409	0.409	1.538
			...		
o_{11}	0.049	0.049	0.622	0.622	1.342
o_{12}	0.360	0.360	0.311	0.311	1.342
			...		
o_{15}	0.049	0.049	0.622	0.622	1.342
o_{16}	0.360	0.360	0.311	0.311	1.342
			...		
o_{27}	0.360	0.360	0.311	0.311	1.342
o_{28}	0.671	0.671	0	0	1.342
			...		
o_{31}	0.360	0.360	0.311	0.311	1.342
o_{32}	0.671	0.671	0	0	1.342

Table 6.8: Exhaustive search for centroids, Weighted Hamming distance.

In this example $m = 5$, $n = 4$ and $l_{max} = 2$. From previous analysis, we know that it takes $(3 \cdot m - 1) \cdot n \cdot (l_{max})^m + (n-1) \cdot (l_{max})^m = (3 \cdot 5 - 1) \cdot 4 \cdot 2^5 + (4-1) \cdot 2^5 = 1888$ basic (like comparison, assignment and addition) operations to perform an exhaustive search over the set O.

Minimal $F_{WH}(o, \hat{E}_c^+(Exp_c))$ is equal to 1.342. There are eight objects $o' \in O$ with that value of $F_{WH}(o, \hat{E}_c^+(Exp_c))$. They constitute a set of candidates for prototype: $Candidates = \{o_{11}, o_{12}, o_{15}, o_{16}, o_{27}, o_{28}, o_{31}, o_{32}\}$, where $o_{11} = [01010]$, $o_{12} = [01011]$, $o_{15} = [01110]$, $o_{16} = [01111]$, $o_{27} = [11010]$, $o_{28} = [11011]$, $o_{31} = [11110]$, $o_{32} = [11111]$.

[4]The values are rounded to three decimals, these rounded values are used in subsequent calculations.

The number of centroids is bigger than in case of unweighted Hamming distance, because information gain and consequently weight of attribute a_3 is 0, so the value of this attribute does not affect the Weighted Hamming distance value.

Optimized extraction of candidates During the first stage of algorithm 6.3 the number of occurrences of values of attributes is calculated (for attributes with $w_a > 0$), and stored in dictionary L_a. The result is shown in table 6.9. It takes $m \cdot l_{max} = 5 \cdot 2 = 10$ assignment operations to initialize dictionary L_a and $n \cdot (m-1) = 4 \cdot 4 = 16$ addition operations to calculate the number of occurrences (additions for attribute a_3 are skipped as its weight is 0).

	a_1	a_2	a_3	a_4	a_5
w_a	0.311	0.049	0	0.049	0.311
$L_a(0)$	2	1	0	1	2
$L_a(1)$	2	3	0	3	2
H_a	{0, 1}	{1}	{0, 1}	{1}	{0, 1}

Table 6.9: Dictionary L_a and most frequent values H_a, Weighted Hamming distance.

In second stage of algorithm 6.3 a set of values v from V_a with the highest number of occurrences is determined for all $a \in A$. The result is also shown in table 6.9. It takes 8 assignment operations to memorize all H_a sets.

During third stage of algorithm 6.3 candidates for prototype are constructed. Since there are eight potential candidates, their construction takes 40 assignment operations. The result is obviously identical to naive approach described above: $Candidates = \{o_{11}, o_{12}, o_{15}, o_{16}, o_{27}, o_{28}, o_{31}, o_{32}\}$, where $o_{11} = [01010]$, $o_{12} = [01011]$, $o_{15} = [01110]$, $o_{16} = [01111]$, $o_{27} = [11010]$, $o_{28} = [11011]$, $o_{31} = [11110]$, $o_{32} = [11111]$. Let us note however, that obtaining this result takes much less basic operations that in case of naive approach – 74 in total, comparing to 1888 in naive approach. Naturally, the difference will be greater as the number of attributes m grows.

Calculation of prototype and quality of model As for the unweighted Hamming distance, the final form of the model strongly depends on which candidate will be accepted first. Of course, the values of the radii and sets $Core/Boundary/Outer$ may be different than for the case of Hamming distance without weights. E.g. for a prototype $o_c^\star = o_{31} = [11110]$:

- $\tau_c^+ = 0.360$,
- $\tau_c^- = 0.622$,
- $Core_c(o_{31}) = \{o_7, o_{13}, o_{28}, o_{32}\}$,

- $Outer_c(o_{31}) = \{o_6, o_8, o_{12}, o_{14}\}$,
- $Boundary_c(o_{31}) = \varnothing$.

The above model can be shown as in the figure 6.3. Obviously, the category is well defined – the model divides above examples into two subsets: belonging and not belonging to the category, thus the border is empty.

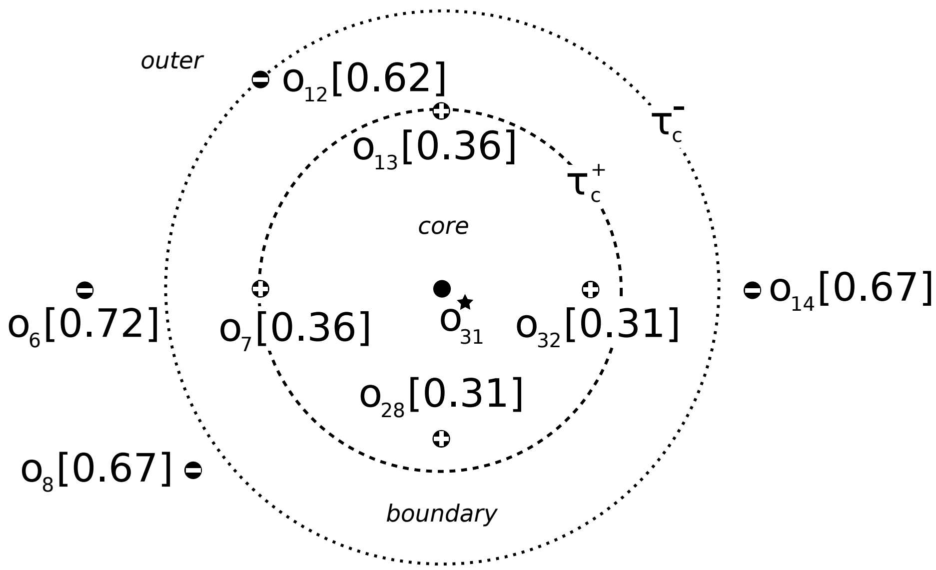


Figure 6.3: Model m_c for Weighted Hamming distance, centroids.

The sets $Core(o_{31})$, $Boundary(o_{31})$ and $Outer(o_{31})$ are identical to the model with unweighted Hamming distance. However, the distances between the objects and the prototype $o_c^\star$ have changed. In addition, for example, the o_{28} object is located slightly closer to the prototype than the o_7 and o_{13} objects. This is due to the different weights of the attributes by which these objects differ from the $o_c^\star$ prototype.

Grounding of statements The grounding of statements does not depend directly on the selection of the *extractCandidates* function, and it is quite similar regardless of the selection of the macrostructure. For the sake of completeness of example, a simple statement grounding for two objects will be shown, while more complicated cases are described in the chapter 5.

We use the same input data as in unweighted Hamming implementation model example (chapter 6.2.3).

episode	object	a_1	a_2	a_3	a_4	a_5	labels
$Episode(t_3)$	$x_{3,1}$	0	0	0	1	1	
	$x_{3,2}$	1	1	1	0	0	

Table 6.10: Example for grounding using Weighted Hamming distance.

The object $x_{3,1}$ from working memory corresponds to object $o_4 = [00011]$ in embodied ontology and the object $x_{3,2}$ corresponds to object $o_{29} = [11100]$. Let us assume that the object $o_{31} = [11110]$ has been chosen as category's prototype $o_c^\star$.

Since $f_H(o_4, o_c^\star) = 0.671 \geqslant \tau_c^- = 0.622$, the object o_4 is located in the outer region of category's model and the epistemic satisfaction relation holds for statement $Know(x_{3,1} \notin c)$ (definition 5.6). Analogously, $f_H(o_{29}, o_c^\star) = 0.049 \leqslant \tau_c^+ = 0.360$, thus the object o_{29} is located in the core of category's model and the epistemic satisfaction relation holds for statement $Know(x_{3,2} \in c)$ (definition 5.5).

6.3.4 Mechanism of medoid candidates extraction

In this chapter, let the *extractCandidates* function determine the medoids of the multiset $\hat{E}_c^+(Exp_c)$ based on the Weighted Hamming distance, more precisely:

Definition 6.8. For a given multiset $\hat{E}_c^+(Exp_c)$ and a macrostructure in a form of Weighted Hamming distance function f_{WH}, an output of the function $extractCandidates$ $(\hat{E}_c^+(Exp_c))$ is defined as a set of such objects o' from E_c^+ that fulfill the condition:

$$F_{WH}(o', \hat{E}_c^+(Exp_c)) = \min_{o \in E_c^+} F_{WH}(o, \hat{E}_c^+(Exp_c)) \tag{6.3.8}$$

that is they minimize the following sum:

$$F_{WH}(o, \hat{E}_c^+(Exp_c)) = \sum_{e \in E_c^+} (k_e^+ \cdot f_{WH}(o, e)). \tag{6.3.9}$$

The definition 6.8 naturally resembles the definition 6.7, as it is still the definition of the central element of $\hat{E}_c^+(Exp_c)$ multiset (given as an input parameter) based on the Weighted Hamming distance. The difference is that the representative is selected from the multiset and not the entire object space O.

The above condition is met by objects $o' \in \hat{E}_c^+(Exp_c)$ for which the sum of Weighted Hamming distances to other objects in the multiset $\hat{E}_c^+(Exp_c)$ is the smallest. This translates into calculating the distance to the remaining[5] elements of the E_c^+ set, taking into account the frequency of occurrence of individual objects in the original $\hat{E}_c^+(Exp_c)$ set. The condition of selecting "from a set" is typical for medoids, and has a clear

[5]Distances to other instances of identical objects may be omitted as they are equal to zero, from the definition of the metric.

practical interpretation here, as it translates into the fact that potential representatives must be objects that have actually been observed by the agent and are present in a relevant part of the learning experience. This translates into the analytical condition $k_{o'}^{+} > 0$.

Due to the fact, that only the multiset $\hat{E}_c^{+}(Exp_c)$ (effectively, only the set E_c^{+} of unique objects of the given multiset) is searched for the central element, the problem of choosing a medoid assuming a naive approach should be less complex task than choosing a centroid.

To find objects o' we can refer directly to the definition 6.8, performing a full search of E_c^{+}, calculating appropriate sum for each $o \in E_c^{+}$.

Theorem 6.22. *Let's assume that $\hat{E}_c^{+}(Exp_c)$ is the given multiset, $m = |A|$ is the cardinality of the set of attributes, and n denotes the cardinality of the set of positive examples for the category c, i.e. $n = |E_c^{+}|$. A full search of E_c^{+} for medoids, i.e. objects that satisfy the condition from the definition 6.8 has the polynomial complexity of $O(m \cdot n^2)$.*

Proof. The calculation of the value of the $f_{WH}(o, e)$ function for the given two objects o and e requires the execution of O(m) basic operations. Since the $f_{WH}(o, e)$ function needs to be computed for each pair of $o, e \in E_c^{+}$ objects, it is necessary to perform $O(m \cdot n^2)$ comparison operations, which means polynomial computational complexity. ■

Despite the polynomial computational complexity of a naive approach, it is worth showing that it can be further reduced. In order to proceed with the reduction it is required to analyze some properties related to medoids evaluated according to definition 6.8. While it is impossible to directly re-apply algorithm 6.3 previously used for an evaluation of centroids, an overall structure of the minimized distance functions remains the same. It is still a sum of Weighted Hamming distances between potential candidates and the remainder of the multiset. Therefore, let us start by repeating equation (6.3.4) which shows a transformed form of the sum which is being minimized, calculated for the candidate o

$$F_{WH}(o, \hat{E}_c^{+}(Exp_c)) = \sum_{a \in A} \left[w_a \cdot \sum_{e \in E_c^{+}} \left(k_e^{+} \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right) \right]. \qquad (6.3.10)$$

Let us notice that for a given $\hat{E}_c^{+}(Exp_c)$ and assumed $a \in A$, an expression in the square brackets of (6.3.10) consists of two factors, the first being the weight w_a, and the second being the sum which evaluates to a number of objects from the learning experience $\hat{E}_c^{+}(Exp_c)$ that exhibit a value **different** than v within the attribute a.

To reformulate this sum, let us re-introduce the following notion describing a number of objects from the learning experience $\hat{E}_c^+(Exp_c)$ that exhibit **the same** value v within the attribute a:

$$\widetilde{S}(v, \hat{E}_c^+(Exp_c), a) = \sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad v \neq e(a) \\ 1 & if \quad v = e(a) \end{cases} \right) \tag{6.3.11}$$

It leads us to the following re-formulation of the previous sum from (6.3.10):

$$\sum_{e \in E_c^+} \left(k_e^+ \cdot \begin{cases} 0 & if \quad o(a) = e(a) \\ 1 & if \quad o(a) \neq e(a) \end{cases} \right) = |\hat{E}_c^+(Exp_c)| - \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a).$$

Using this equality for the substitution in (6.3.10) we get the following:

$$F_{WH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[w_a \cdot \left(|\hat{E}_c^+(Exp_c)| - \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right) \right] =$$

$$= \sum_{a \in A} \left[w_a \cdot |\hat{E}_c^+(Exp_c)| \right] - \sum_{a \in A} \left[w_a \cdot \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right],$$

where the component

$$\sum_{a \in A} \left[w_a \cdot |\hat{E}_c^+(Exp_c)| \right] \tag{6.3.12}$$

is constant for a given $\hat{E}_c^+(Exp_c)|$ and assumed distance function (Weighted Hamming distance with pre-defined weights).

It entails that during an optimization (**minimization**) of $F_{WH}(o, \hat{E}_c^+(Exp_c))$, the whole component (6.3.12) can be omitted, which leads to an equivalent minimization of the following expression:

$$-\sum_{a \in A} \left[w_a \cdot \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right],$$

which is, in turn, equivalent to a **maximization** of the following proposed criterion in its final form:

$$\widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \left[w_a \cdot \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a) \right]. \tag{6.3.13}$$

It further leads to a formulation of the following lemma:

Theorem 6.23. *For a given multiset $\hat{E}_c^+(Exp_c)$ and a macrostructure in a form of Weighted Hamming distance function f_{WH}, objects o' satisfy definition 6.8 if and only* *if they satisfy the following condition*

$$\widetilde{F}_{WH}(o', \hat{E}_c^+(Exp_c)) = \max_{o \in E_c^+} \widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c)). \tag{6.3.14}$$

Proof. A proof consists of the whole already-presented deduction preceding the theorem. ■

Remark 6.2. *If a weight w_a for a given attribute $a \in A$ is equal to 0, then a respective component*

$$w_a \cdot \widetilde{S}(o(a), \hat{E}_c^+(Exp_c), a)$$

is also reduced to 0. This property could be incorporated into the following algorithm in a similar fashion to how it was done in algorithm 6.3. However, since it would complicate the algorithm without reducing an overall computational complexity, we will present it in the form provided below.

This leads to the formulation of a practical three-phase algorithm 6.4. In the first phase (lines 1-5), for each attribute, the frequency of individual values of that attribute is calculated. In the second phase (lines 6-12), these frequencies are used to determine the value of the optimization criterion (at the same time, the maximum value of the criterion is determined – lines 11-12). The third phase (lines 13-16) constructs the target set of solutions.

Remark 6.3. *The third stage (lines 13-16) of the algorithm could be incorporated into the second stage (lines 6-12) but it could cause excessive juggling of a set Candidates which would need to be purged as many times as the maximum value of the criterion is replaced in line 12.*

Let us proceed with an evaluation of a computational complexity of algorithm 6.4 in the presented form. It is easy to notice that algorithm 6.4 differs from algorithm 6.2 only in line 10, where the criterion $\widetilde{F}_{WH}$ is additionally influenced by weights w_a. It does not change the overall computational complexity, leading to the folowing result (without repeating the proof):

Theorem 6.24. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – the cardinality of the set of positive examples for the category c, $l_{max} = \max_{a \in A} |V_a|$ – an overestimation of the cardinality of sets V_a for $a \in A$. A computational complexity of algorithm 6.4 is polynomial $O(m \cdot (l_{max} + n))$.*

In practical applications $(l_{max} + n)$ component should be much lower than n^2 – e.g. in case where all attributes are binary $l_{max} = 2$, and $2 + n < n^2$ for all $n > 2$. Therefore computational complexity of algorithm 6.4 should be lower accordingly in

Algorithm 6.4: *extractCandidates* – evaluation of medoids for distance f_{WH}

Input: a multiset $\hat{E}_c^+(Exp_c)$, a distance function f_{WH}, a set of weights w_a for all $a \in A$

Output: set of candidates $Candidates$

foreach *attribute a in A* **do**
  Initialize a dictionary $L_a = \{\langle v, 0 \rangle : \; v \in V_a\}$;
foreach *object o in* E_c^+ **do**
  foreach *attribute a in A* **do**
    $L_a(o(a)) \leftarrow L_a(o(a)) + k_o^+$;
Initialize a 'maximal' value of a criterion as zero: $\widetilde{F}_{WH,max} \leftarrow 0$;
foreach *object o in* E_c^+ **do**
  Initialize a criterion for a current object: $\widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c)) = 0$;
  foreach *attribute a in A* **do**
    $\widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c)) \leftarrow \widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c)) + w_a \cdot L_a(o(a))$;
  if $\widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c)) > \widetilde{F}_{WH,max}$ **then**
    $\widetilde{F}_{WH,max} \leftarrow \widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c))$
Initialize $Candidates \leftarrow \varnothing$;
foreach *object o in* E_c^+ **do**
  if $\widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c)) == \widetilde{F}_{WH,max}$ **then**
    Add o to a set $Candidates$;

comparison to the naive approach. This reduction of computational complexity from $O(m \cdot (l_{max} + n))$ to $O(m \cdot n^2))$ may have an influence on a fluent behavior of the system for large sets of agent's learning experience.

It is worth noticing that in particular implementations many of the weighted sums naively computed in presented algorithms using *for* loops could be implemented in a more effective way using hardware supporting fast matrix-based operations. However, we leave an evaluation of the complexity in the provided general form.

6.3.5 Illustrative example – medoids

Input data in this example are identical like in example using centroid candidates extraction method - they are shown in table 6.7. Therefore, in this chapter only key differences will be shown.

Initial computations Like in previous example: $Exp_c = \{1(o_7, +)^c, 1(o_{13}, +)^c, 1(o_{28}, +)^c, 1(o_{32}, +)^c, 1(o_6, -)^c, 1(o_8, -)^c, 1(o_{12}, -)^c, 1(o_{14}, -)^c\}$ where $o_6 = [00101]$, $o_7 = [00110]$, $o_8 = [00111]$, $o_{12} = [01011]$, $o_{13} = [01100]$, $o_{14} = [01101]$, $o_{28} = [11011]$, $o_{32} = [11111]$.

Also:

- $E^+ = E_c^+(Exp_c) = \{o_7, o_{13}, o_{28}, o_{32}\}$,
- $E^- = E_c^-(Exp_c) = \{o_6, o_8, o_{12}, o_{14}\}$.

Naive extraction of candidates Again, for this example, the information gain (i.e. the weight of a given attribute) for the following attributes is calculated as follows:

- $w_{a_1} = Gain(Exp_c, a_1) = 0.311$,
- $w_{a_2} = Gain(Exp_c, a_2) = 0.049$,
- $w_{a_3} = Gain(Exp_c, a_3) = 0$,
- $w_{a_4} = Gain(Exp_c, a_4) = 0.049$,
- $w_{a_5} = Gain(Exp_c, a_5) = 0.311$.

In order to find candidates by naively following the definition 6.8, one has to perform an exhaustive search over the set E_c^+, involving calculation of the sum (6.3.9) for every potential candidate $o \in E_c^+$. Data produced during that process are shown in table 6.11.

$o \in O$	$f_{WH}(o, o_7)$	$f_{WH}(o, o_{13})$	$f_{WH}(o, o_{28})$	$f_{WH}(o, o_{32})$	$F_{WH}(o, \hat{E}_c^+(Exp_c))$
o_7	0	0.098	0.671	0.671	1.440
o_{13}	0.098	0	0.671	0.671	1.440
o_{28}	0.671	0.671	0	0	1.342
o_{32}	0.671	0.671	0	0	1.342

Table 6.11: Exhaustive search for medoids, Weighted Hamming distance.

In this example $m = 5$ and $n = 4$. From previous analysis, we know that it takes $(3 \cdot m - 1) \cdot n^2 + (n-1) \cdot n = (3 \cdot 5 - 1) \cdot 4^2 + (4-1) \cdot 4 = 236$ basic (like comparison, assignment and addition) operations to perform an exhaustive search over the set E_c^+. Let us note, that it is much lower number than in case of naive search for centroids, since in that example the set E_c^+ is much smaller than O.

Minimal $F_{WH}(o, \hat{E}_c^+(Exp_c))$ is equal to 1.342. There are two objects $o' \in E_c^+$ with that value of $F_{WH}(o, \hat{E}_c^+(Exp_c))$. They constitute a set of candidates for prototype: $Candidates = \{o_{28}, o_{32}\}$. The number of medoids is bigger than in case of unweighted Hamming distance, because information gain and consequently weight of attribute a_3 is 0, so the value of this attribute does not affect the Weighted Hamming distance value.

Optimized extraction of candidates During the first stage of algorithm 6.4 the number of occurrences of values of attributes is calculated, and stored in dictionary L_a (similar to centroid case). The result is shown in table 6.12. It takes $m \cdot l_{max} = 5 \cdot 2 = 10$

assignment operations to initialize dictionary L_a and $n \cdot m = 4 \cdot 5 = 20$ addition operations to calculate number of occurrences.

	a_1	a_2	a_3	a_4	a_5
w_a	0.311	0.049	0	0.049	0.311
$L_a(0)$	2	1	1	1	2
$L_a(1)$	2	3	3	3	2

Table 6.12: Dictionary L_a, Weighted Hamming distance.

In the second stage of algorithm 6.4 optimization criterion $\widetilde{F}_{WH}(o, \hat{E}_c^+(Exp_c))$ is calculated for each object $o \in E_c^+$. The result is:

- $\widetilde{F}_{WH}(o_7, \hat{E}_c^+(Exp_c)) = 1.440$
- $\widetilde{F}_{WH}(o_{13}, \hat{E}_c^+(Exp_c)) = 1.440$
- $\widetilde{F}_{WH}(o_{28}, \hat{E}_c^+(Exp_c)) = 1.538$
- $\widetilde{F}_{WH}(o_{32}, \hat{E}_c^+(Exp_c)) = 1.538$.

from that $\widetilde{F}_{WH,max} = 1.538$. It takes $n \cdot m = 4 \cdot 5 = 20$ addition and also 20 multiplication operations to calculate optimization criteria.

During the third stage of algorithm 6.4 candidates for prototype are chosen. It takes $n = 4$ comparation operations to find them. The result is obviously identical to naive approach described above: $Candidates = \{o_{28}, o_{32}\}$. Obtaining this result takes less basic operations that in case of naive approach – 74 in total, comparing to 236 in naive approach. Let us note however, that this example is very small, for the sake of simplicity. Naturally, there will be much bigger difference in favor of optimized approach if n is bigger, which should be the case in practical applications.

Calculation of prototype and quality of model Let us assume, that candidate $o_{32} = [11111]$ is first to be checked by algorithm 4.2, then:

- $\tau_c^+ = 0$,
- $\tau_c^- = NULL$,
- $Core_c(o_{32}) = \{o_{28}, o_{32}\}$,
- $Outer_c(o_{32}) = \varnothing$,
- $Boundary_c(o_{32}) = \{o_6, o_7, o_8, o_{12}, o_{13}, o_{14}\}$.

The acceptance condition is fulfilled, because $|Core_c(o_{32})| = 2 \geqslant 2 = |Boundary_c(o_{32}) \cap E^+|$. In consequence, o_{32} will be set as a prototype of category c. This model is shown on figure 6.4. The τ_c^+ radius is exaggerated on the figure, because it is actually equal to 0.

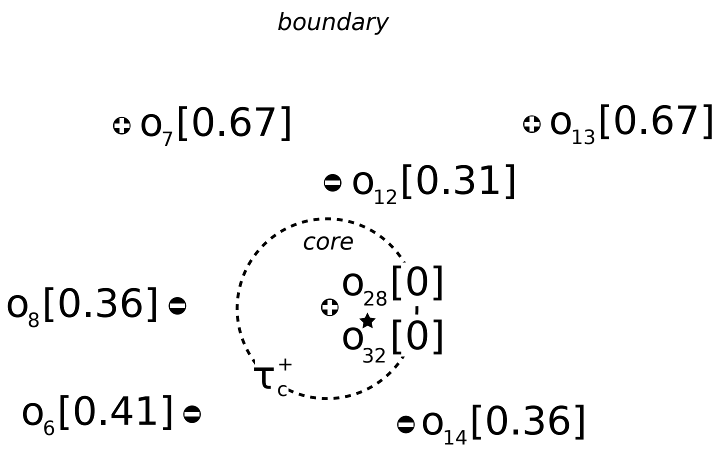


Figure 6.4: Model m_c for Weighted Hamming distance, medoids.

Clearly, in this example the set of candidates is much smaller in medoids case, than in centroids case. As a result, the quality of model in medoids case may be worse than in centorids case, depending on which of the centroid and medoid candidates will be chosen. On the other hand, searching for medoid candidates is less computationally expensive, especially in case of naive search.

6.4 Asymmetric Weighted Hamming distance

As in the previous chapter, the assumption for this model is to meet the conditions (a) and (b) from 6.1.1, i.e. to find objects that have the most properties in common with other elements of the category, and at the same time the least properties in common with elements of the opposite categories. Ideas in this chapter are also based on processing model proposed by Rosch, which uses notion of **cue validity**[75]:

"Cue validity is a probabilistic concept; the validity of a given cue x as a predictor of a given category y (the conditional probability $y|x$) increases as the frequency with which cue x is associated with category y increases and decreases as the frequency with which cue x is associated with category other than y increases[4, 5, 69]. (...)The cue validity of an entire category may be defined as the summation of the cue validities for that category of each of the attributes of the category. (...)A category with high cue validity is, by definition, more differentiated from other categories than one of lower

cue validity.”

Rosch argues[74] that in modeling of natural categories, prototypes and cue validity should be incorporated into a single model, as they appear to be the basis of categories. Such model would be efficient processing mechanism of categorization, because matching to a prototype wouldn't need to use much of processing power for calculating cue validities. Rosch states basing on research and experiments concerning human cognition:

“In short, humans probably incorporate probabilistic analysis of cues and computation of distance from a representation of the category into the same process of categorization;”

At first glance, it would seem that we could incorporate above ideas to be used with Weighted Hamming distance, by assigning conditional probabilities to weights of the attributes describing objects. The problem is, that conditional probabilities vary for different values of attributes. Even in case of binary attributes, usually conditional probability $P(y|x) \neq P(y|\neg x)$. Thus we need to extend the granularity of definition of Weighted Hamming distance, so it associates the weights (conditional probabilities) with particular values of attributes, not the attribute as a whole. Such measure is going to be called Asymmetric Weighted Hamming distance (AWH).

6.4.1 Macrostructure

Definition 6.9. The Asymmetric Weighted Hamming distance f_{AWH} for the vectors o_i and o_j is defined as follows:

$$f_{AWH}(o_i, o_j) = \sum_{a \in A} \begin{cases} 0 & if \quad o_i(a) = o_j(a) \\ w_{o_j(a)} & if \quad o_i(a) \neq o_j(a). \end{cases}$$

where $o_j(a) \in V_a$ is a value of attribute a for object o_j, and thus $w_{o_j(a)} \geqslant 0$ denotes a weight of value $v \in V_a$ equal to $o_j(a)$.

Property 6.4. For any two objects $o_i, o_j \in O$ evaluation of Asymmetric Weighted Hamming distance $f_{AWH}(o_i, o_j)$ requires $m = |A|$ comparison operations, m assignments, and *m-1* additions, giving an overall computational complexity of $O(m + m + m - 1) = O(m)$.

Let us note, that this function is not symmetrical, i.e. $f_{AWH}(o_i, o_j)$ does not have to be equal to $f_{AWH}(o_j, o_i)$. This might be counter-intuitive, but it is consistent with the results of psychological experiments[41] where there is also asymmetry, e.g. in generalization.

Commonsense interpretation of this measure is that it represents distance from object o_i to object o_j. Naturally, if for these objects the values of some attribute a are different, the distance is interpreted as greater, than if the values are the same. This is also the case for symmetrical Hamming distances described in previous chapters. But Asymmetrical Weighted Hamming distance takes into account, that specific values of attribute a for target object o_j may have varying importance, expressed numerically as weight of this value. It may be also interpreted as an "effort" of changing the value of attribute a in object o_i to the value present in object o_j – the bigger is the weight of target value, the bigger is the "effort".

Let us consider an example, where objects are described with two attributes: *color* and *shape*. The domains of above attributes are:

- $V_{color} = \{white, gray, black\}$
- $V_{shape} = \{circle, square\}$.

The values of weights are given as:

- $w_{white} = 0.8, w_{gray} = 0.2, w_{black} = 0.8$
- $w_{circle} = 1.0, w_{square} = 0.0$.

Let us assume that white circle is a target object o_j. The figure 6.5 depicts distances $f_{AWH}(o_i, o_j)$ of various objects o_i from object o_j. The dotted circles are radii that show specific distance to object o_j, note that they are not drawn in scale.

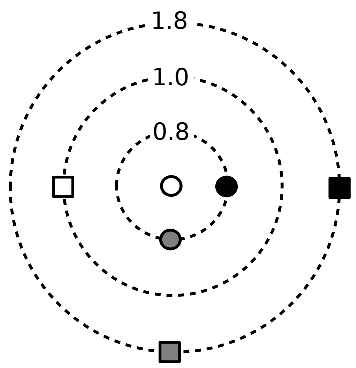


Figure 6.5: Example of f_{AWH} distances calculation.

The measure f_{AWH} forms a space in which objects that have many attribute's values with high weights common with target object are closest to it. The measure f_{AWH} is "focused" on target object, because it takes into account only its weights of values of attributes. From that, for example in the figure 6.5 the gray and black circles are in the same distance from target object, despite the fact that $w_{black} = 0.8$ is much higher that $w_{gray} = 0.2$. The common-sense interpretation of that effect is, that weights represent importance of particular value of attribute in objects that possess that value. So when calculating the distance of these two circles to target object o_j, the important factor is that they are not white, and not their actual color.

Concordant to Rosch's proposal, the weights of the values of the attributes will be equal to their cue validities. Cue validity modeled on [74] is defined as the frequency of the attribute value associated with a given category, divided by the total frequency of this attribute value for all other relevant categories. Technically, it translates to the definition, where cue validity is the conditional probability $P(c|a = v)$ that tells us how good the c category predictor is the value of v of the attribute a.

Definition 6.10. Let $s^+(c, v)$ be the number of positive learning experiences of the c category, that have the value of v for the attribute a, so:

$$s^+(c, v) = |\{o \in E_c^+ : o(a) = v\}|$$

Let $s(c, v)$ be the number of all learning experiences of the c category, that have the value of v for the attribute a, so:

$$s(c, v) = |\{o \in Exp_c : o(a) = v\}|$$

Then the conditional probability $P(c|a = v)$ of belonging to the category c provided that the value of the a attribute is v, is defined as:

$$P(c|a = v) = \frac{s^+(c, v)}{s(c, v)}.$$

This definition is an extension of the idea from work [34]. As can be seen, it relates to the agent's empirical experience and does not take into account the multiplicities k_o^+ and k_o^- of object observations. The consequence is, that the observation of multiple identical objects (or multiple observations of the same object) affect the cue validity value as much as one such observation. It seems reasonable, as for example a hundred observations of the same bird that has wings, should not increase the cue validity of value *yes* of attribute *has wings* for category *bird*, the same as observations of a hundred different birds with wings. Rosch herself does not give any guidance on this matter, as she does not try to approximate the value of cue validities in human processing. If one wants to include more statistical information in cue validity, one has to take into account multiplicities k_o^+ and k_o^-. There are also different types of measures of the association between a feature and a category present in literature that can be used, like collocation or category utility, e.g. see [99].

Property 6.5. Assume $m = |A|$ – a number of attributes, $n = |Exp_c|$ – a cardinality of the set of examples for the category c.

Computing the set of cue validities (conditional probabilities) for all the values of all the attributes has a computational complexity of $O(m \cdot n)$.

As already mentioned, in this model we equate AWH weights with conditional probabilities, so for category c for all values $v \in V_a$ of attributes $a \in A$ the weights are defined as $w_v = P(c|a = v)$. Therefore, it is possible to rewrite the definition 6.9 as:

$$f_{AWH}(o_i, o_j) = \sum_{a \in A} \begin{cases} 0 & if \quad o_i(a) = o_j(a) \\ P(c|a = o_j(a)) & if \quad o_i(a) \neq o_j(a). \end{cases}$$

Example 5. Let us assume a microstructure of all objects o from a cognitive universe O in a form of fixed-length binary vectors, where length $n = 3$ and two example objects: $o_1 = [001]$ and $o_2 = [010]$. The set of attributes is denoted as $A = (a_1, a_2, a_3)$. Let the probabilities $P(c|a = v)$ be given:

	a_1	a_2	a_3
$P(c\|a_i = 0)$	0.3	0.8	0.5
$P(c\|a_i = 1)$	1.0	0.4	1.0

Table 6.13: Probabilities for example of f_{AWH} calculation.

The result of calculations is as follows:

$$f_{AWH}(o_1, o_2) = 0 + P(c|a_2 = 1) + P(c|a_3 = 0) = 0 + 0.4 + 0.5 = 0.9$$

$$f_{AWH}(o_2, o_1) = 0 + P(c|a_2 = 0) + P(c|a_3 = 1) = 0 + 0.8 + 1.0 = 1.8$$

6.4.2 Mechanism of centroid candidates extraction

In this chapter, let the *extractCandidates* function determine the centroids of the multiset $\hat{E}_c^+(Exp_c)$ based on maximal sum of weigths, more precisely:

Definition 6.11. Given a set of weights associated with values of attributes, such that w_v is a weight of value $v \in V_a$, an output of $extractCandidates(\hat{E}_c^+(Exp_c))$ is defined as a set of such $o' \in O$ that fulfill the condition:

$$F_{AWH}(o', \hat{E}_c^+(Exp_c)) = \max_{o \in O} F_{AWH}(o, \hat{E}_c^+(Exp_c)) \tag{6.4.1}$$

that is they maximize the following sum:

$$F_{AWH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \sum_{\substack{e \in \hat{E}_c^+ \\ \wedge \\ e(a) = o(a)}} (k_e^+ \cdot w_{o(a)}). \tag{6.4.2}$$

Again, taking into account that in this model weights are equal to cue validities, equation (6.4.2) can be rewritten as:

$$F_{AWH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a)=o(a)}} (k_e^+ \cdot P(c|a = o(a)).$$

The above condition is fulfilled by such objects o' that maximize sum of cue validities on all attributes in reference to $\hat{E}_c^+(Exp_c)$ multiset. This criterion is derived from model proposed by Rosch[76], where it is stated that cue validity of an entire category may be defined as the summation of the cue validities for that category of each of the properties of the category. Obviously it is different approach, that in case of symmetrical Hamming functions, where sum of distances to objects from $\hat{E}_c^+(Exp_c)$ is minimized. The latter type of criteria could be investigated in future research. In that case, interesting question arises: whether to minimize a sum of distances **from** other objects or **to** other objects, which is crucial provided that asymmetrical distance function is used.

Let us note, that in above definition the multiplicity of object observations is taken into account. This is not stated in any of Rosch's works, but seems to be necessary to avoid undesirable effect that attribute value with high cue validity but very rarely observed, is included in a prototype. Let us consider a situation depicted in figure 6.6: 7 out of 10 white objects belong to the category, so cue validity of value *white* for attribute *color* is equal to $7/10 = 0.7$. There is also only one black object, and it does belong to the category, so cue validity of value *black* for attribute *color* is equal to $1/1 = 1$. Thus, if we would only choose the features with highest cue validity, the black color would be chosen as a feature of category's prototype, when in fact it is rather uncommon than representative for that category. However since in model proposed in this work the multiplicity is taken into account, which is expressed as component $(k_e^+ \cdot w_{o(a)})$ in equation (6.4.2), the value *white* would be chosen, as $(k_e^+ \cdot w_{white}) = 7 \cdot 0.7 = 4.9$ and $(k_e^+ \cdot w_{black}) = 1 \cdot 1 = 1$.

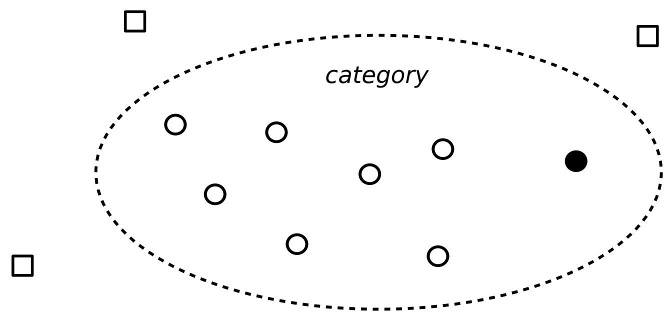


Figure 6.6: Cue validity calculation example.

Nevertheless, the objects fulfilling condition (6.4.1) are centroids, since they represent multiset $\hat{E}_c^+(Exp_c)$ and (as opposed to the medoid determination task, which is analyzed in later chapters) they are selected from all available space. This means that for selected objects it may happen that the multiplicity of $k_{o'}^+$ is 0, which is equivalent to the fact that for a given o' object there is no positive learning material about such an object and a given concept. In order to find such objects o' one can naively follow the definition 6.11 in a direct way by performing an exhaustive search over the set O, and by calculating the sum (6.4.2) for every potential candidate $o \in O$.

Theorem 6.25. *Naive approach, that is, an exhaustive search over the set O in order to find objects satisfying the condition (6.4.1) formulated in definition 6.11 leads to an exponential computational complexity in relation to a number of attributes used for describing objects from O provided that a number of values of each attribute is non-trivial ($\geqslant 2$) and finite.*

Proof. Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c. To estimate the complexity from below, let us assume that all attributes $a \in A$ are binary (that is, they have exactly two distinct values each). Then a cardinality of the universe $|O| = 2^m$. A computational complexity of each calculation of the expression $(k_e^+ \cdot w_{o(a)})$ is $O(1)$. This expression needs to be evaluated for each attribute of (at most) each object $e \in E_c^+$, which leads to $O(m \cdot n)$ multiplication and addition operations. The above calculations result in the value of $F_{AWH}(o, \hat{E}_c^+(Exp_c))$, which has to be evaluated for each $o \in O$. It leads to a final polynomial-exponential computational complexity $O(m \cdot n \cdot 2^m)$ with an exponential component dependent on m.

Naturally, as a number of values of any attribute $a \in A$ is greater than 2, a cardinality of the universe O grows, which translates to a higher number of required evaluations of $F_{AWH}(o, \hat{E}_c^+(Exp_c))$ and, in consequence, to higher complexity (e.g., if every attribute as exactly three distinct values, the final complexity becomes $O(m \cdot n \cdot 3^m)$).

In general, a component 2^m is a multiplication of cardinalities of sets of values of respective properties. To estimate the complexity from above, let us assume that l_{max} is the largest cardinality of sets of properties' values (it exists since there is always a maximum integer in a finite set of finite integers). It leads to an approximation of the upper bound of the complexity as $O(m \cdot n \cdot (l_{max})^m)$, which is still exponential in relation to m. ∎

The use of exponential complexity algorithms in a practical context is very limiting and often imposes strict restrictions on the size of the problem. In particular, it would be necessary to provide input data instances containing only a small number of attributes.

It is easy to notice that for a given $\hat{E}_c^+(Exp_c)$, for an assumed a, and provided that values of weights $w_{o(a)}$ are finite and greater than *0*, a value of the expression on the right side of equation (6.4.2), that is,

$$\sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a)=o(a)}} (k_e^+ \cdot w_{o(a)}), \tag{6.4.3}$$

depends only on a choice of $o(a)$ and, what is more important, it does not depend on values of other attributes, both within object o, and within other objects e.

In terms of interpretation, an expression (6.4.3) aggregates an overall influence related to the attribute a on the final value of F_{AWH}. It means that when choosing from the whole universe O, that is, when all combinations of values are allowed, we can optimize (6.4.3) for every attribute separately. Each particular $w_{o(a)}$ is constant. Let us introduce the notion, given that $v \in V_a$ is one of values of the attribute a:

$$S_A(v, \hat{E}_c^+(Exp_c), a) = \sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a)=v}} (k_e^+ \cdot w_v) = w_v \cdot \sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a)=v}} k_e^+$$

which makes the following lemma convenient to formulate:

Lemma 6.26. *A necessary and sufficient condition for an object $o' = (o'(a_1), o'(a_2), \ldots, o'(a_m))$ to be a proper centroid (satisfying (6.4.1)) is to fulfill a set of sub-conditions:*

$$S_A(o'(a_i), \hat{E}_c^+(Exp_c), a_i) = \max_{v \in V_{a_i}} S_A(v, \hat{E}_c^+(Exp_c), a_i) \tag{6.4.4}$$

for $i = 1, 2, \ldots, m$.

Proof. It follows directly from an argumentation preceding the lemma. ∎

From lemma 6.26 we can directly derive procedure for construction of the candidate. We need to simply assign such a value v to attribute a, that maximizes $S_A(v, \hat{E}_c^+(Exp_c), a)$. If there is more than one such value, then any of them is equally good value of attribute a, thus we need to construct multiple candidates with all values v of a, that maximize $S_A(v, \hat{E}_c^+(Exp_c), a)$. Three-phase algorithm 6.5 implements above ideas. In the first phase (lines 1-8) it calculates a sum of weights of particular values within respective attributes, that is values of $S_A(v, \hat{E}_c^+(Exp_c), a)$. In the second phase (lines 9-10) the best values are chosen for each attribute (depending on calculated sum $S_A(v, \hat{E}_c^+(Exp_c), a)$). In the third phase (lines 11-20) a set of solutions (here: candidates) is constructed.

Algorithm 6.5: *extractCandidates* for a set of weights associated with values of attributes

Input: a multiset $\hat{E}_c^+(Exp_c)$, a set of weights w_v for all $v \in V_a$ for all $a \in A$
Output: set of candidates $Candidates$

1 **foreach** *attribute* a *in* A **do**
2   Initiate a dictionary $L_a = \{\langle v, 0\rangle : \; v \in V_a\}$;
3 **foreach** *object* o *in* E_c^+ **do**
4   **foreach** *attribute* a *in* A **do**
5         $L_a(o(a)) \leftarrow L_a(o(a)) + k_o^+$;
6 **foreach** *attribute* a *in* A **do**
7   **foreach** *value* v *in* V_a **do**
8         $L_a(v) \leftarrow L_a(v) \cdot w_v$;
9 **foreach** *attribute* a *in* A **do**
10     $H_a \leftarrow$ a set of values v from V_a with the highest value of $L_a(v)$;
11 $o \leftarrow$ new object;
12 $Candidates \leftarrow \{o\}$;
13 **foreach** *attribute* a *in* A **do**
14     $newCandidates \leftarrow \varnothing$;
15   **foreach** *value* h *in* H_a **do**
16   **foreach** *object* o *in* *Candidates* **do**
17             $o' \leftarrow$ duplicate of o;
18             $o'(a) \leftarrow h$;
19   add o' to $newCandidates$;
20     $Candidates \leftarrow newCandidates$;

As stated in theorem 6.25, a naive approach to evaluation of a centroid of a multiset has a mixed polynomial-exponential complexity between $O(m \cdot n \cdot 2^m)$ and $O(m \cdot n \cdot (l_{max})^m)$. Let us show the reduced computational complexity provided by algorithm 6.5.

Lemma 6.27. *A computational complexity of the first stage (lines 1-8) of algorithm 6.5 is approximated from above by* $O(m \cdot l_{max}) + O(n \cdot m) + O(m \cdot l_{max}) = O(m \cdot (l_{max} + n))$.

Proof. Initialization of dictionary L_a (lines 1-2) has to be performed for each of m attributes. In the worst-case scenario, the number of values for an attribute is assumed to be equal to l_{max}, thus computational complexity of lines 1-2 is $O(m \cdot l_{max})$.

Calculation of frequency of values in objects in E_c^+ (lines 3-5) involves two loops - iterating over n objects and m attributes. Thus computational complexity of lines 3-5 is $O(n \cdot m)$.

In lines 6-8 the above frequencies are multiplied by weights in order to get values of $S_A(v, \hat{E}_c^+(Exp_c), a)$. It involves two loops - iterating over m attributes and at most l_{max} values of respective attributes. Thus computational complexity of lines 6-8 is $O(m \cdot l_{max})$.

It lets us evaluate an overall complexity of lines 1-8 as $O(m \cdot l_{max}) + O(n \cdot m) + O(m \cdot l_{max}) = O(m \cdot (l_{max} + n))$. ■

Lemma 6.28. *A computational complexity of the second stage (lines 9-10) of algorithm 6.5 is $O(m \cdot l_{max})$.*

Proof. A loop in line 9 iterates m times. The choice of the highest values is approximated as $O(l_{max})$ – their further assignment to H_a does not further raise a character of this complexity. It lets us evaluate an overall computational complexity of lines 9-10 as $O(m \cdot l_{max})$. ■

A complexity of the third stage (lines 11-20) of algorithm 6.5 requires a more extended comment. It is easy to notice that a set of optimal solutions (following the notions used in the algorithm) takes a form of the following Cartesian product:

$$Candidates = H_{a_1} \times H_{a_2} \times \cdots \times H_{a_m} \tag{6.4.5}$$

and its size is directly dependent on a cardinality of particular sets H_a. It means that in a borderline case where there is exactly one most common value per attribute, there is exactly one optimal solution. However, as a number of 'ties' grows within particular attributes, a set *Candidates* grows exponentially in size (per each attribute with more than one optimal value). It leads to an exponential evaluation of the worst-case complexity for the third stage of the algorithm (note that in our assumptions all optimal solutions need to be listed and returned in their final form; were it instead required to provide just one optimal solution, it would become as simple as $O(m)$).

Lemma 6.29. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c. As elements of sets H_a are chosen from corresponding sets V_a, then their cardinality cannot be larger than their respective counterparts. In consequence, $|H_a| \leqslant |V_a|$ for $a \in A$, and therefore it is possible to estimate the cardinality of all H_a from above by $l_{max} = \max_{a \in A} |V_a|$. A computational* *complexity of a third stage (lines 11-20) of algorithm 6.5 is approximated from above by $O(l_{max}^m)$ for the worst-case scenario.*

Proof. The complexity follows directly from a need to generate a whole Cartesian product (6.4.5) and an estimation mentioned in the lemma itself. It leads to the following dependencies:

$$\prod_{a \in A} |H_a| \leqslant \prod_{a \in A} |V_a| \leqslant \prod_{a \in A} \left(\max_{a' \in A} |V_{a'}| \right) = \prod_{a \in A} l_{max} = l_{max}^m.$$

■

A computational complexity of the worst-case scenario, as already mentioned, is extremely higher in comparison to the complexity of the best-case scenario which is summed up within a following straightforward lemma.

Lemma 6.30. *Assume $m = |A|$ – a number of attributes. If every H_a contains just a single element, that is if there is a single most frequent value for each attribute $a \in A$, then a computational complexity of the third stage (lines 11-20) of algorithm 6.5 is polynomial, $O(m)$.*

Proof. If there is a single most frequent value for each attribute $a \in A$, then $Candidates$ set includes only one element through all iterations of algorithm. The construction of that element needs m assignment operations. Thus computational complexity of lines 11-20 is $O(m)$. ■

It leads to the final estimation of a computational complexity of the whole algorithm 6.5.

Theorem 6.31. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c, $l_{max} = \max_{a \in A} |V_a|$ – an overestimation of the cardinality of sets V_a for $a \in A$.*

A computational complexity of the whole algorithm in the best-case scenario (with only one optimal candidate) is not worse than $O(m \cdot (l_{max} + n) + m \cdot l_{max} + m) = O(m \cdot (l_{max} + n))$.

A computational complexity of the whole algorithm in the worst-case scenario (with all potential candidates being optimal) is not worse than $O(m \cdot (l_{max} + n) + m \cdot l_{max} + l_{max}^m) = O(m \cdot (l_{max} + n) + l_{max}^m)$.

Proof. A proof follows directly from previous evaluations of computational complexities of particular parts which are simply sequentially connected in algorithm 6.5. ■

Conclusions. Two extreme situations regarding the computational complexity of the algorithm 6.5 are presented above. In practical applications, it is unlikely that many attributes have more than one most common value, so the computational complexity of the algorithm should approach the polynomial. The algorithm 6.5 can therefore be used in practice, which was confirmed by the simulation tests presented in chapter 7.

Worth noting is the fact, that compared to the naive approach, characterized by absolutely exponential computational complexity, the obtained algorithm reviews the original learning material (the multiset $\hat{E}_c^+(Exp_c)$) only once, and a potential exponential complexity element occurs only in the pessimistic case, when the size of the set of optimal solutions is very numerous and is related only to the construction of all instances of optimal candidates, not to their determination.

In a particular version of the algorithm 4.2, where only one optimal candidate would be searched and not the whole set of optimal candidates, the computational complexity of the proposed computational approach would be strictly polynomial.

6.4.3 Illustrative example – centroids

Let us assume, for the sake of simplicity, that the agent's experience consists of only two episodes. The world accessible to an agent's perception is described by four attributes. Three of them are binary (a_1, a_2, a_3), one is ternary (a_4). Agent's observations collected during two episodes are presented in the table below. As can be seen, they concern only one category c.

episode	**object**	a_1	a_2	a_3	a_4	**labels**
$Episode(t_1)$	$x_{1,1}$	0	0	0	0	*is-c*
	$x_{1,2}$	1	0	1	0	*is-c*
	$x_{1,3}$	1	0	0	2	*not-c*
	$x_{1,4}$	0	0	1	2	*not-c*
$Episode(t_2)$	$x_{2,1}$	0	1	0	0	*is-c*
	$x_{2,2}$	0	0	1	1	*is-c*
	$x_{2,3}$	1	1	0	0	*not-c*
	$x_{2,4}$	1	0	1	2	*not-c*

Table 6.14: Example using Asymmetric Weighted Hamming distance.

Initial computations Based on the above observations, as a result of the preprocessing performed by the agent after the second episode, its experience regarding the category c will be collected in the form of a multiset $Exp_c = \{1(o_1,+)^c, 1(o_5,+)^c, 1(o_7,+)^c, 1(o_{16},+)^c, 1(o_6,-)^c, 1(o_{15},-)^c, 1(o_{18},-)^c, 1(o_{19},-)^c\}$ where $o_1 = [0000]$, $o_5 = [0011]$, $o_6 = [0012]$, $o_7 = [0100]$, $o_{15} = [1002]$, $o_{16} = [1010]$, $o_{18} = [1012]$, $o_{19} = [1100]$. Note, that $O = \{o_1, ..., o_{24}\}$.

It follows from the above:

- $E^+ = E_c^+(Exp_c) = \{o_1, o_5, o_7, o_{16}\}$,
- $E^- = E_c^-(Exp_c) = \{o_6, o_{15}, o_{18}, o_{19}\}$.

Based on the data from the table 6.14, the following probability values can be calculated:

	a_1	a_2	a_3	a_4
$P(c\|a_i = 0)$	0.75	0.5	0.5	0.75
$P(c\|a_i = 1)$	0.25	0.5	0.5	1.0
$P(c\|a_i = 2)$				0.0

Table 6.15: Probabilities for example.

They also constitute the weights for all respective values $v \in V_a$.

Naive extraction of candidates In order to find candidates by naively following the definition 6.11, one has to perform an exhaustive search over the set O, involving calculation of the sum (6.4.2) for every potential candidate $o \in O$. Most important data produced during that process are shown in table 6.16, naturally complete table consists of 24 rows.

$o \in O$	$F_{AWH}(o, \hat{E}_c^+(Exp_c))$
o_1	7
o_2	5.75
o_3	4.75
o_4	7
	...
o_{23}	2.75
o_{24}	1.75

Table 6.16: Exhaustive search for centroids, Asymmetric Weighted Hamming distance.

In this example $m = 4$, $n = 4$ and $l_{max} = 3$. Number of basic (like multiplication, comparison and addition) operations to perform an exhaustive search over a set O, depends on values of objects in E^+ (only for objects $o \in O$ that have the same value of attribute the multiplication and addition is performed). Without going into more details, it takes 352 basic operations to perform an exhaustive search over the set O in this example.

Maximal $F_{AWH}(o, \hat{E}_c^+(Exp_c))$ is equal to 7. There are two objects $o' \in O$ with that value of $F_{AWH}(o, \hat{E}_c^+(Exp_c))$. They constitute a set of candidates for prototype: $Candidates = \{o_1, o_4\}$, where $o_1 = [0000]$, $o_4 = [0010]$.

Optimized extraction of candidates During the first stage of algorithm 6.5 values of $S_A(v, \hat{E}_c^+(Exp_c), a)$ are calculated, and stored in dictionary L_a. The result is shown in table 6.17. It takes 9 assignment operations to initialize dictionary L_a and $n \cdot m = 4 \cdot 4 = 16$ addition operations to calculate number of occurrences.

	a_1	a_2	a_3	a_4
$L_a(0)$	2.25	1.5	1	2.25
$L_a(1)$	0.25	0.5	1	1
$L_a(2)$				0
H_a	{0}	{0}	{0, 1}	{0}

Table 6.17: Dictionary L_a and most frequent values H_a, Asymmetric Weighted Hamming distance.

In the second stage of algorithm 6.5 a set H_a of values v from V_a with the highest value of $S_A(v, \hat{E}_c^+(Exp_c), a)$ is determined for all $a \in A$. The result is also shown in table 6.17. It takes 5 assignment operations to memorize all H_a sets.

During the third stage of algorithm 6.5 candidates for prototype are constructed. Since there are two potential candidates, their construction takes 8 assignment operations. The result is obviously identical to naive approach described above: $Candidates = \{o_1, o_4\}$, where $o_1 = [0000]$, $o_4 = [0010]$. Let us note however, that obtaining this result takes much less basic operations than in case of naive approach – 38 in total, comparing to 352 in naive approach. Naturally, the difference will be greater as the number of attributes m grows.

Calculation of prototype and quality of model The quality of the model of category strongly depends on the order in which the indicators for the candidates are calculated. If vector $o_1 = [0000]$ is selected first, then the model coefficients look like this:

- $\tau_c^+ = 0.5$,
- $\tau_c^- = 1.5$,
- $Core_c(o_1) = \{o_1, o_7\}$,
- $Outer_c(o_1) = \{o_{15}, o_{18}\}$,
- $Boundary_c(o_1) = \{o_5, o_6, o_{16}, o_{19}\}$.

In this case, the results achieved meet the condition for category c to be learned because $|Core_c(o_1)| = 2 \geqslant 2 = |Boundary_c(o_1) \cap E^+|$. In consequence, the object o_1 is assigned as the prototype $o_c^\star$ of the properly established category c with τ_c^+ and τ_c^- as its radii. The category c can be integrated with the ontological knowledge base.

The above model can be shown as in the figure 6.7. As we can see, the category is well defined, although the boundary contains four objects. The distances of objects to the prototype are shown in square brackets.

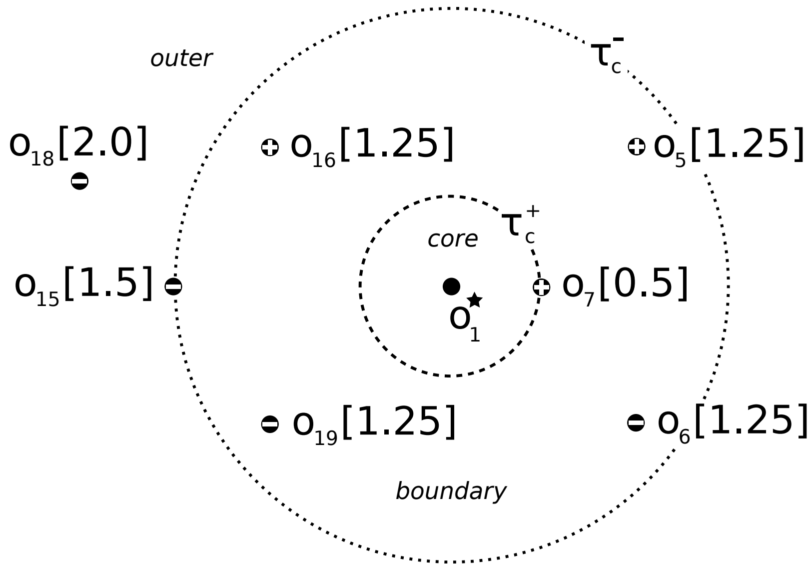


Figure 6.7: Model m_c for Asymmetric Weighted Hamming distance.

Grounding of statements The grounding of statements does not depend directly on the selection of the *extractCandidates* function, and it is quite similar regardless of the selection of the macrostructure. For the sake of completeness of example, a simple statement grounding for two objects will be shown, while more complicated cases are described in the chapter 5.

Let us assume, that in $Episode(t_3)$ two objects are available to agent's perception – their description is given in table 6.18. The objects are not marked with any labels by the teacher.

episode	**object**	a_1	a_2	a_3	a_4	**labels**
$Episode(t_3)$	$x_{3,1}$	0	0	1	0	
	$x_{3,2}$	1	1	0	2	

Table 6.18: Example for grounding using Asymmetric Weighted Hamming distance.

The object $x_{3,1}$ from working memory corresponds to object $o_4 = [0010]$ in embodied ontology and the object $x_{3,2}$ corresponds to object $o_{21} = [1102]$. Let us assume that the object $o_1 = [0000]$ has been chosen as category's prototype $o_c^\star$.

Since $f_{AWH}(o_{21}, o_c^\star) = 2.0 \geqslant \tau_c^- = 1.5$, the object o_{21} is located in the outer region of category's model and the epistemic satisfaction relation holds for statement $Know(x_{3,1} \notin c)$ (definition 5.6). Analogously, $f_{AWH}(o_4, o_c^\star) = 0.5 \leqslant \tau_c^+ = 0.5$, thus the object o_4 is located in the core of category's model and the epistemic satisfaction relation holds for statement $Know(x_{3,2} \in c)$ (definition 5.5).

6.4.4 Mechanism of medoid candidates extraction

In this chapter, let the *extractCandidates* function determine the medoids of the multiset $\hat{E}_c^+(Exp_c)$ based on the sum of weights, more precisely:

Definition 6.12. Given a set of weights associated with values of attributes, such that w_v is a weight of value $v \in V_a$, an output of $extractCandidates(\hat{E}_c^+(Exp_c))$ is defined as a set of such $o' \in E_c^+$ that fulfill a condition:

$$F_{AWH}(o', \hat{E}_c^+(Exp_c)) = \max_{o \in E_c^+} F_{AWH}(o, \hat{E}_c^+(Exp_c)) \tag{6.4.6}$$

that is they maximize the following sum:

$$F_{AWH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a) = o(a)}} (k_e^+ \cdot w_{o(a)}). \tag{6.4.7}$$

The definition 6.12 naturally resembles the definition 6.11, as it is still the definition of the representative element of $\hat{E}_c^+(Exp_c)$ multiset (given as an input parameter) based on the set of weights associated with values of attributes. The difference is that the representative is selected from the multiset and not the entire object space O.

Again, taking into account that in this model weights are equal to cue validities, equation (6.4.7) can be rewritten as:

$$F_{AWH}(o, \hat{E}_c^+(Exp_c)) = \sum_{a \in A} \sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a) = o(a)}} (k_e^+ \cdot P(c|a = o(a)).$$

The above condition is met by objects $o' \in \hat{E}_c^+(Exp_c)$ for which the sum of cue validities in the multiset $\hat{E}_c^+(Exp_c)$ is the greatest. The condition of selecting "from a set" is typical for medoids, and has a clear practical interpretation here, as it translates into the fact that potential representatives must be objects that have actually been observed by the agent and are present in a relevant part of the learning experience. This translates into an analytical condition $k_{o'}^+ > 0$.

Due to the fact, that only the multiset $\hat{E}_c^+(Exp_c)$ (effectively, only the set E_c^+ of unique objects of the given multiset) is searched for the representative element, the problem of choosing a medoid assuming a naive approach should be less complex task than choosing a centroid.

To find objects o' we can refer directly to the definition 6.12, performing a full search of E_c^+, counting appropriate sum for each $o \in E_c^+$.

Theorem 6.32. *Let's assume that $\hat{E}_c^+(Exp_c)$ is the given multiset, $m = |A|$ is the cardinality of the set of attributes, and n denotes the cardinality of the set of positive examples for the category c, i.e. $n = |E_c^+|$. A full search of E_c^+ for medoids, i.e. objects that satisfy the condition from the definition 6.12 has the polynomial complexity of $O(m \cdot n^2)$.*

Proof. A computational complexity of each calculation of the expression $(k_e^+ \cdot w_{o(a)})$ is $O(1)$. This expression needs to be evaluated for each attribute of (at most) each object $e \in E_c^+$, which leads to $O(m \cdot n)$ multiplication and addition operations. The above calculations result in the value of $F_{AWH}(o, \hat{E}_c^+(Exp_c))$, which has to be evaluated for each $o \in E_c^+$. It leads to a final polynomial computational complexity $O(m \cdot n^2)$. ■

Despite a polynomial computational complexity of a naive approach, it is worth showing that it can be further reduced. While it is impossible to directly re-apply algorithm 6.5 previously used for an evaluation of centroids, it is possible to avoid repetitions of calculations on the right side of equation (6.4.7). Let us note, that for a given $o(a)$ the inner sum is the same for all objects $o' \in E_c^+$ that have the same value of attribute a, i.e. for which $o(a) = o'(a)$. The above sum is actually $S_A(v, \hat{E}_c^+(Exp_c), a)$ introduced in previous chapter:

$$S_A(v, \hat{E}_c^+(Exp_c), a) = \sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a)=v}} (k_e^+ \cdot w_v) = w_v \cdot \sum_{\substack{e \in E_c^+ \\ \wedge \\ e(a)=v}} k_e^+.$$

The values of $S_A(v, \hat{E}_c^+(Exp_c), a)$ can be memorized for all values $v \in V_a$ and used for faster calculation of $F_{AWH}(o, \hat{E}_c^+(Exp_c))$.

This leads to the formulation of a practical three-phase algorithm 6.6. In the first phase (lines 1-8) it calculates sum of cue validities of particular values within respective attributes, that is values of $S_A(v, \hat{E}_c^+(Exp_c), a)$. In the second phase (lines 9-15), these sums are used to determine the value of the optimization criterion (at the same time, the maximum value of the criterion is determined – lines 14-15). The third phase (lines 16-19) constructs the target set of solutions.

Remark 6.4. *The third stage (lines 16-19) of the algorithm could be incorporated into the second stage (lines 9-15) but it could cause excessive juggling of a set Candidates which would need to be purged as many times as the maximum value of the criterion is replaced in line 15.*

Algorithm 6.6: *extractCandidates* – evaluation of medoids for a set of weights associated with values of attributes

Input: a multiset $\hat{E}_c^+(Exp_c)$, a set of weights w_v for all $v \in V_a$ for all $a \in A$
Output: set of candidates $Candidates$

1 **foreach** *attribute* a *in* A **do**
2 Initiate a dictionary $L_a = \{\langle v, 0\rangle :\ v \in V_a\}$;
3 **foreach** *object* o *in* E_c^+ **do**
4 **foreach** *attribute* a *in* A **do**
5 $L_a(o(a)) \leftarrow L_a(o(a)) + k_o^+$;
6 **foreach** *attribute* a *in* A **do**
7 **foreach** *value* v *in* V_a **do**
8 $L_a(v) \leftarrow L_a(v) \cdot w_v$;
9 Initialize the 'maximal' value of a criterion as zero: $F_{AWH,max} \leftarrow 0$;
10 **foreach** *object* o *in* E_c^+ **do**
11 Initialize a criterion for a current object: $F_{AWH}(o, \hat{E}_c^+(Exp_c)) = 0$;
12 **foreach** *attribute* a *in* A **do**
13 $F_{AWH}(o, \hat{E}_c^+(Exp_c)) \leftarrow F_{AWH}(o, \hat{E}_c^+(Exp_c)) + L_a(o(a))$;
14 **if** $F_{AWH}(o, \hat{E}_c^+(Exp_c)) > F_{AWH,max}$ **then**
15 $F_{AWH,max} \leftarrow F_{AWH}(o, \hat{E}_c^+(Exp_c))$
16 Initialize $Candidates \leftarrow \varnothing$;
17 **foreach** *object* o *in* E_c^+ **do**
18 **if** $F_{AWH}(o, \hat{E}_c^+(Exp_c)) == F_{AWH,max}$ **then**
19 Add o to a set $Candidates$;

Let us proceed with an evaluation of a computational complexity of algorithm 6.6 in the presented form. The first stage of the algorithm is identical to the first stage of algorithm 6.5, so its complexity is presented without repeating the proof.

Lemma 6.33. *A computational complexity of the first stage (lines 1-8) of algorithm 6.6 is approximated from above by* $O(m \cdot l_{max}) + O(n \cdot m) + O(m \cdot l_{max}) = O(m \cdot (l_{max} + n))$.

Lemma 6.34. *Assume* $m = |A|$ *– a number of attributes,* $n = |E_c^+|$ *– a cardinality of the set of positive examples for the category* c*. A computational complexity of the second stage of algorithm 6.6 (lines 9-15) is polynomial* $O(n \cdot m)$.

Proof. Line 9 has a complexity of $O(1)$. The loop in lines 10-15 is run n times. Inside that loop in line 11 we have variable initialization of complexity $O(1)$, in lines 12-13 there is an inner loop that executes m times (inside this loop, on line 13, a variable is incremented by the value of another variable, $O(1)$), and on lines 14-15 a conditional statement and an optional assignment operation (also $O(1)$). The computational complexity of the entire block (lines 9-15) can therefore be estimated at the level of $O(1 + n \cdot (1 + (m \cdot 1) + 1 \cdot 1)) = O(n \cdot m)$. ■

Lemma 6.35. *Let n denote the cardinality of the set of unique positive examples for the category c (that is $n = |E_c^+|$). The computational complexity of the third phase of the algorithm 6.6 (lines 16-19) is linear $O(n)$.*

Proof. Line 16 contains only initialization. In lines 17-19, the *for* loop is executed n times. Inside loop there is a comparison $O(1)$ and possibly addition of object to the set of solutions $O(1)$. So the complexity can be estimated at $O(1+n\cdot(1+1)) = O(n)$. ■

Theorem 6.36. *Assume $m = |A|$ – a number of attributes, $n = |E_c^+|$ – a cardinality of the set of positive examples for the category c, $l_{max} = \max_{a\in A} |V_a|$ – an overestimation of the cardinality of sets V_a for $a \in A$. A computational complexity of Algorithm 6.6 is polynomial $O(m \cdot (l_{max} + n))$.*

Proof. A proof follows directly from previous evaluations of computational complexities of particular parts which are simply sequentially connected in algorithm 6.6: $O(m \cdot (l_{max} + n)) + O(n \cdot m) + O(n) = O(m \cdot (l_{max} + n))$. ■

In practical applications $(l_{max} + n)$ component should be much lower than n^2 – e.g. in case where all attributes are binary $l_{max} = 2$, and $2 + n < n^2$ for all $n > 2$. Therefore computational complexity of algorithm 6.6 should be lower accordingly in comparison to the naive approach. This reduction of computational complexity from $O(m\cdot(l_{max}+n))$ to $O(m\cdot n^2))$ may have an influence on a fluent behavior of the system for large sets of agent's learning experience.

6.4.5 Illustrative example – medoids

Input data in this example are identical like in example using centroid candidates extraction method – they are shown in table 6.14. Therefore, in this chapter only key differences will be shown.

Initial computations Like in the previous example: $Exp_c = \{1(o_1,+)^c$, $1(o_5,+)^c$, $1(o_7,+)^c$, $1(o_{16},+)^c$, $1(o_6,-)^c, 1(o_{15},-)^c, 1(o_{18},-)^c, 1(o_{19},-)^c\}$ where $o_1 = [0000]$, $o_5 = [0011]$, $o_6 = [0012]$, $o_7 = [0100]$, $o_{15} = [1002]$, $o_{16} = [1010]$, $o_{18} = [1012]$, $o_{19} = [1100]$. Note, that $O = \{o_1, ..., o_{24}\}$.

Also:

- $E^+ = E_c^+(Exp_c) = \{o_1, o_5, o_7, o_{16}\}$,
- $E^- = E_c^-(Exp_c) = \{o_6, o_{15}, o_{18}, o_{19}\}$.

Probability values that constitute the weights for all respective values $v \in V_a$:

	a_1	a_2	a_3	a_4
$P(c\|a_i = 0)$	0.75	0.5	0.5	0.75
$P(c\|a_i = 1)$	0.25	0.5	0.5	1.0
$P(c\|a_i = 2)$				0.0

Table 6.19: Probabilities for example, Asymmetric Weighted Hamming distance.

Naive extraction of candidates In order to find candidates by naively following the definition 6.12, one has to perform an exhaustive search over a set E_c^+, involving calculation of a sum (6.4.7) for every potential candidate $o \in E_c^+$. Data produced during that process are shown in table 6.20.

$o \in O$	$F_{AWH}(o, \hat{E}_c^+(Exp_c))$
o_1	7
o_5	5.75
o_7	6
o_{16}	5

Table 6.20: Exhaustive search for medoids, Asymmetric Weighted Hamming distance.

In this example $m = 4$, $n = 4$ and $l_{max} = 3$. Number of basic (like multiplication, comparison and addition) operations to perform an exhaustive search over a set E_c^+, depends on values of objects in E^+ (only for objects $o \in E_c^+$ that have the same value of attribute the multiplication and addition is performed). Without going into more details, it takes 76 basic operations to perform an exhaustive search over a set E_c^+ in this example. Let us note, that it is much lower number than in case of naive search for centroids, since in that example the set E_c^+ is much smaller than O.

Maximal $F_{AWH}(o, \hat{E}_c^+(Exp_c))$ is equal to 7. There is one object $o' \in E_c^+$ with that value of $F_{AWH}(o, \hat{E}_c^+(Exp_c))$. It is therefore the only member of a set of candidates for prototype: $Candidates = \{o_1\}$.

Optimized extraction of candidates During the first stage of algorithm 6.6 values of $S_A(v, \hat{E}_c^+(Exp_c), a)$ are calculated, and stored in dictionary L_a. The result is shown in table 6.21. It takes 9 assignment operations to initialize dictionary L_a and $n \cdot m = 4 \cdot 4 = 16$ addition operations to calculate number of occurrences.

	a_1	a_2	a_3	a_4
$L_a(0)$	2.25	1.5	1	2.25
$L_a(1)$	0.25	0.5	1	1
$L_a(2)$				0

Table 6.21: Dictionary L_a, Asymmetric Weighted Hamming distance.

In the second stage of algorithm 6.6 optimization criterion $F_{AWH}(o, \hat{E}_c^+(Exp_c))$ is calculated for each object $o \in E_c^+$. The result was already shown in table 6.20. It is obviously identical to naive extraction, but it can be calculated using lower number of operations. It takes $n \cdot (m-1) = 4 \cdot 3 = 12$ addition operations to calculate optimization criteria.

From above $F_{AWH,max} = 7$, it takes 4 comparison operations to find it.

During the third stage of algorithm 6.6 candidates for prototype are chosen. It takes $n = 4$ comparation operations to find them. The result is obviously identical to naive approach described above: $Candidates = \{o_1\}$. Obtaining this result takes less basic operations that in case of naive approach – 41 in total, comparing to 76 in naive approach. Let us note however, that this example is very small, for the sake of simplicity. Naturally, there will be much bigger difference in favor of optimized approach if n is bigger, which should be the case in practical applications.

Calculation of prototype and quality of model The set of candidates is smaller than in case of centroids extraction. Thus, it is possible that the quality of prototype would be lower. But in example for centroids extraction, the object o_1 was also chosen as a prototype, so the result is identical. For details please refer to chapter 6.4.3.

Grounding of statements Again, the results are identical as in case of centroids. They are described in chapter 6.4.3.

Chapter 7

Simulation studies

The models presented above have been implemented in a computer system, which ultimately proves their technical feasibility. This chapter mainly presents the results of simulation of the learning strategy for different models, i.e. the quality and time of learning the embodied ontology. Also presented are the exemplary statements generated by the agent, using the described cognitive semantics. Deeper analysis of the grounded formulas and communication interaction would require a more advanced implementation of the agent, e.g. placing it in a physical environment in a robot's body.

The data sets used in the simulation studies were taken from the Machine Learning Repository provided by the University of California, Irvine.

7.1 Database for fitting contact lenses

The data set „Database for fitting contact lenses" was downloaded from https://archive.ics.uci.edu/ml/datasets/lenses. The data set contains 24 examples (objects), divided into 3 classes:

- *hard* – the patient should be fitted with hard contact lenses,
- *soft* – the patient should be fitted with soft contact lenses,
- *none* – the patient should not be fitted with contact lenses.

The patients are described by four attributes – one of them is ternary, the others are binary:

- age of the patient: presbyopic(2), young(1), pre-presbyopic(0),
- spectacle prescription: myope(1), hypermetrope(0),
- astigmatic: no(1), yes(0),
- tear production rate: reduced(1), normal(0).

7.1.1 Hamming distance

Category	*Candidates*	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
none	[2001]	none	1	4	6	2	16
soft	[0010],[1010]	none	0	3	1	9	14
hard	[1100]	none	0	2	1	18	5

Table 7.1: Lens dataset, Hamming distance, centroids.

Centroid version The table 7.1 shows for each of the categories the set of candidates returned by the *extractCandidates* function and (possibly) the prototype selected for the model. As one can see, none of the candidates for all three categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned. For this reason, the values of the radii and sizes of individual model areas presented in the table refer to the prototype candidate (in case of *soft* category to [1010]). One can notice a relatively small size of the cores and large boundaries, which suggests the low quality of the obtained models.

Since we are dealing with three ill-defined categories, the agent will not generate any statements about membership of the objects to the above categories.

Category	*Candidates*	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
none	[2001]	none	1	4	6	2	16
soft	[0010],[1010]	none	0	3	1	9	14
hard	[1100]	none	0	2	1	18	5

Table 7.2: Lens dataset, Hamming distance, medoids.

Medoid version The results are identical to the case with centroids.

7.1.2 Weighted Hamming distance

Category	*Candidates*	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
none	[2001]	[2001]	0.55	0.59	13	6	5
soft	[0010],[1010]	[0010]	0.02	0.25	4	18	2
hard	[1100]	[1100]	0.04	0.05	4	20	0

Table 7.3: Lens dataset, Weighted Hamming distance, centroids.

Centroid version The table 7.3 shows for each of the categories the set of candidates returned by the *extractCandidates* function and the prototype selected for the model. As can be seen, prototypes have been found for all three categories because the quality condition has been met for them. The values of the radii and the size of the individual areas of the model presented in the table refer to the prototype. One can notice a relatively small size of the boundary (for the *hard* category it is even empty), which suggests the high quality of the model obtained. The result is therefore much better than with the unweighted Hamming distance.

The prototype patients in each category have the following characteristics:

- *none*: age=presbyopic, spectacle=hypermetrope, astigmatic=yes, tear=reduced,
- *soft*: age= pre-presbyopic, spectacle=hypermetrope, astigmatic=no, tear=normal,
- *hard*: age=young, spectacle=myope, astigmatic=yes, tear=normal.

It can be seen that the candidate sets are identical to the case of Hamming distances without weights. Whereas the introduction of weights changed the structure of the mental space, thanks to which it was possible to better distinguish elements belonging and not belonging to the category, as well as to select a prototype.

Cognitive structures created in the learning process on the basis of the above data can be used by the agent to generate statements in accordance with the assumed cognitive semantics. The examples of valid grounded formulas are presented below:

- for the object $x = [1111]$ the epistemic satisfaction relation holds for formula $Know(x \in none)$, because the corresponding object o is at the distance of 0.04 from prototype, so in the core of model of category *none*,
- for the same object $x = [1111]$ the epistemic satisfaction relation holds for formula $Know(x \notin hard)$, because the corresponding object o is at the distance of 0.38 from prototype, so in the outer region of model of category *hard*,
- for the object $x = [0000]$ the corresponding object o lies at the distance of 0.58 from the prototype, i.e. within the boundary of model of category *none*; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{none} = 0.019$; for this size of the epistemic neighborhood $EN_{none}(o, \varepsilon_{none})$ of the object o, the relative grounding strength $\lambda_{none}(EN_{none}(o, \varepsilon_{none})) = 0.25$; hence for $\lambda_{minBel} = 0.5$ the epistemic satisfaction relation holds for formulas $Bel(x \notin none)$ and $Pos(x \in none)$,
- for the object $x = [0110]$ the corresponding object o lies at the distance of 0.024 from the prototype, i.e. within the boundary of model of category $soft$; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{soft} = 0.116$; for this size of the epistemic neighborhood $EN_{soft}(o, \varepsilon_{soft})$ of the object o, the relative grounding strength o equals $\lambda_{soft}(EN_{soft}(o, \varepsilon_{soft})) = 0.83$; hence for $\lambda_{minBel} = 0.5$ the epistemic satisfaction relation holds for formulas $Bel(x \in soft)$ and $Pos(x \notin soft)$.

Category	*Candidates*	**Prototype**	τ_c^+	τ_c^-	$\lvert Core\rvert$	$\lvert Outer\rvert$	$\lvert Boundary\rvert$
none	[2001]	[2001]	0.55	0.59	13	6	5
soft	[0010],[1010]	[1010]	0.02	0.25	4	18	2
hard	[1100]	[1100]	0.04	0.05	4	20	0

Table 7.4: Lens dataset, Weighted Hamming distance, medoids.

Medoid version The results are almost identical to the centroid version. The only difference is that a different prototype for the category $soft$ has been chosen. This is accidental outcome, as the candidates are evaluated in a random order. However, it has not changed the quality of the model at all.

7.1.3 Asymmetric Weighted Hamming distance

Category	*Candidates*	**Prototype**	τ_c^+	τ_c^-	$\lvert Core\rvert$	$\lvert Outer\rvert$	$\lvert Boundary\rvert$
none	[2001]	[2001]	1.42	2.42	11	6	7
soft	[0010],[1010]	[1010]	0.25	0.67	4	16	4
hard	[1100]	[1100]	0.25	0.33	4	20	0

Table 7.5: Lens dataset, Asymmetric Weighted Hamming distance, centroids.

Centroid version The table 7.5 shows for each of the categories the set of candidates returned by the *extractCandidates* function and the prototype selected for the model. As can be seen, prototypes have been found for all three categories because the quality condition has been met for them. The values of the radii and the size of the individual areas of the model presented in the table refer to the prototype. One can notice a relatively small size of the boundary (for the *hard* category it is even empty), which suggests the high quality of the model obtained. The result is therefore much better than with the unweighted Hamming distance.

The candidates sets and prototypes are identical to the ones found with symmetric Weighted Hamming distance. However, the radii, *Core*, *Outer* and *Boundary* sets are different.

Cognitive structures created in the learning process on the basis of the above data can be used by the agent to generate statements in accordance with the assumed cognitive semantics. The examples of valid grounded formulas are presented below:

- for the object $x = [1110]$ the epistemic satisfaction relation holds for formula $Know(x \in soft)$, because the corresponding object o is at the distance of 0.25 from prototype, so in the core of model of category $soft$,
- for the same object $x = [1110]$ the epistemic satisfaction relation holds for formula $Know(x \notin none)$, because the corresponding object o is at the distance of 3.08 from prototype, so in the outer region of model of category $none$,

- for the object $x = [0110]$ the corresponding object o lies at the distance of 0.50 from the prototype, i.e. within the boundary of model of category $soft$; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{soft} = 0.21$; for this size of the epistemic neighborhood $EN_{soft}(o, \varepsilon_{soft})$ of the object o, the relative grounding strength is $\lambda_{soft}(EN_{soft}(o, \varepsilon_{soft})) = 0.13$; hence for $\lambda_{minBel} = 0.5$ epistemic satisfaction relation holds for formulas $Bel(x \notin soft)$ and $Pos(x \in soft)$,

- for the object $x = [0111]$ the corresponding object o lies at the distance of 2.08 from the prototype, i.e. within the boundary of model of category $none$; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{none} = 0.5$; for this size of the epistemic neighborhood $EN_{none}(o, \varepsilon_{none})$ of the object o, the relative grounding strength is $\lambda_{none}(EN_{none}(o, \varepsilon_{none})) = 0.67$; hence for $\lambda_{minBel} = 0.5$ the epistemic satisfaction relation holds for formulas $Bel(x \in none)$ and $Pos(x \notin none)$.

Category	*Candidates*	**Prototype**	τ_c^+	τ_c^-	$\lvert Core\rvert$	$\lvert Outer\rvert$	$\lvert Boundary\rvert$
none	[2001]	[2001]	1.42	2.42	11	6	7
soft	[0010],[1010]	[1010]	0.25	0.67	4	16	4
hard	[1100]	[1100]	0.25	0.33	4	20	0

Table 7.6: Lens dataset, Asymmetric Weighted Hamming distance, medoids.

Medoid version The results are identical to the case with centroids.

7.2 Primary Tumor Domain

The data set „Primary Tumor Domain” was downloaded from https://archive.ics.uci.edu/ml/datasets/Primary+Tumor. This primary tumor domain was obtained from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia. Thanks go to M. Zwitter and M. Soklic for providing the data.

The data set contains 339 examples (objects), divided into 22 classes (categories) representing patients suffering from respective tumors. As some categories are very few in number, the top three were selected:

- *lung* – 84 examples,
- *stomach* – 39 examples,
- *ovary* – 29 examples.

Patients are described by 17 attributes – three of them are ternary, the others are binary:

- age: $>= 60(2)$, $< 30(1)$, $30 - 59(0)$,
- sex: male(1), female(0),
- histologic-type: anaplastic(2), epidermoid(1), adeno(0),
- degree-of-diffe: poorly(2), well(1), fairly(0),
- bone: yes(1), no(0),
- bone-marrow: yes(1), no(0),
- lung: yes(1), no(0),
- pleura: yes(1), no(0),
- peritoneum: yes(1), no(0),
- liver: yes(1), no(0),
- brain: yes(1), no(0),
- skin: yes(1), no(0),
- neck: yes(1), no(0),
- supraclavicular: yes(1), no(0),
- axillar: yes(1), no(0),
- mediastinum: yes(1), no(0),
- abdominal: yes(1), no(0).

The data set contains examples with unknown values, which are not suitable for learning phase of the agent. They were removed, leaving 57 examples, which are realizations of 51 individual mental objects.

The size of the universe of mental representations of distinguishable objects O is $3^3 \cdot 2^{14} = 442368$ objects, which is a considerable number. Thus, it is possible to compare calculation time for naive and optimized algorithms of candidates extraction. The calculation times will be shown in the following chapters. Simulations were conducted on computer with Intel(R) Core(TM) i5-4310U CPU @ 2.00-2.60 GHz and 8GB RAM. The algorithms were implemented using C$^{\#}$/.NET framework. Let us note, that calculation times are not very accurate, since they were measured in multitasking operation system (MS Windows 10) and not in real time operating system. Also, .NET background processes like garbage collector can affect the performance. Still, the times are accurate enough to show differences between implemented algorithms.

7.2.1 Hamming distance

Category	$\lvert Candidates \rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
lung	4	none	2	7	6	4	41
stomach	1	none	1	12	0	0	51
ovary	1	none	0	6	1	16	34

Table 7.7: Tumor dataset, Hamming distance, centroids.

Centroid version The table 7.7 shows for each of the categories the size of the set of candidates returned by the $extractCandidates$ function and the prototype selected for the model. As one can see, none of the candidates for all three categories has met the qualitative condition, therefore no prototype has been selected and the categories were not learned. For this reason, the values of the radii and sizes of individual model areas presented in the table refer to the prototype candidate. One can notice a relatively small size of the cores and large boundaries, which suggests the low quality of the obtained models.

Since all three categories are ill-defined, the agent does not generate any statements about membership of the objects to the above categories.

Total time of calculations for extraction of candidates for all three models, using naive procedure was 21556 ms, and using optimized procedure it was 5 ms.

Category	$\lvert Candidates \rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
lung	1	none	1	8	2	2	47
stomach	1	none	1	NULL	1	0	50
ovary	1	none	0	6	1	16	34

Table 7.8: Tumor dataset, Hamming distance, medoids.

Medoid version The results are shown in table 7.8, they are similar to the case with centroids. Again, none of the candidates for all three categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned.

Total time of calculations for extraction of candidates for all three models, using naive procedure was 4 ms, and using optimized procedure it was 4 ms.

7.2.2 Weighted Hamming distance

Category	$\lvert Candidates \rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
lung	8	[01121000 000000110]	0.58	1.13	15	18	18
stomach	4	none	NULL	0.34	0	18	33
ovary	4	none	0	0.09	3	40	8

Table 7.9: Tumor dataset, Weighted Hamming distance, centroids.

Centroid version The table 7.9 shows for each of the categories the size of the set of candidates returned by the *extractCandidates* function and the prototype selected for the model. As can be seen, a prototype has been found for one of three categories. The values of the radii and the size of the individual areas of the model presented in the table refer to the prototype, or to the candidate if the category is ill-defined. One can notice smaller size of the boundary of respective models of categories, than with the unweighted Hamming distance, which suggests the higher quality of the model obtained.

The prototypical patient for category *lung* has the following characteristics: age=30-59, sex=male, histologic-type=epidermoid, degree-of-diffe=poorly, bone=yes, bone - marrow=no, lung=no, pleura=no, peritoneum=no, liver=no, brain=no, skin=no, neck =no, supraclavicular=no, axillar=yes, mediastinum=yes, abdominal=no.

Total time of calculations for extraction of candidates for all three models, using naive procedure was 22978 ms, and using optimized procedure it was 13 ms.

Cognitive structures for *lung* category, created in the learning process on the basis of the above data, can be used by the agent to generate statements in accordance with the assumed cognitive semantics. The examples of valid grounded formulas are presented below:

- for the object x = [01111000000000000] the epistemic satisfaction relation holds for formula $Know(x \in lung)$, because the corresponding object o is at the distance of 0.39 from the prototype, so in the core of model of category *lung*,
- for the object x = [11010000100001000] the epistemic satisfaction relation holds for formula $Know(x \notin lung)$, because the corresponding object o is at the distance of 1.34 from the prototype, so in the outer region of model of category *lung*,
- for the object x = [01021000000100000] the corresponding object o lies at the distance of 0.71 from the prototype, i.e. within the boundary of model of category *lung*; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{lung} = 0.274$; for this size of the epistemic

neighborhood $EN_{lung}(o, \varepsilon_{lung})$ of the object o, the relative grounding strength is $\lambda_{lung}(EN_{lung}(o, \varepsilon_{lung})) = 0.40$; hence for $\lambda_{minBel} = 0.5$ the epistemic satisfaction relation holds for formulas $Bel(x \notin lung)$ and $Pos(x \in lung)$,

- for the object $x = [11021110011000111]$ the corresponding object o lies at the distance of 0.63 from the prototype, i.e. within the boundary of model of category $lung$; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{lung} = 0.274$; for this size of the epistemic neighborhood $EN_{lung}(o, \varepsilon_{lung})$ of the object o, the relative grounding strength is $\lambda_{lung}(EN_{lung}(o, \varepsilon_{lung})) = 0.60$; hence for $\lambda_{minBel} = 0.5$ epistemic satisfaction relation holds for formulas $Bel(x \in lung)$ and $Pos(x \notin lung)$.

Category	$\lvert Candidates\rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core\rvert$	$\lvert Outer\rvert$	$\lvert Boundary\rvert$
lung	1	none	0.43	1.27	13	9	29
stomach	1	none	NULL	0.34	0	18	33
ovary	1	none	0	0.09	3	40	8

Table 7.10: Tumor dataset, Weighted Hamming distance, medoids.

Medoid version The results are identical to the centroid version for *stomach* and *ovary* categories. However, for *lung* category there was only one candidate, while in centroids case there were eight candidates. Unfortunately, the only candidate has not met the acceptance criteria, so it could not be assigned as a prototype. Thus the category *lung* is ill-defined.

Since all three categories are ill-defined, the agent does not generate any statements about membership of the objects to the above categories.

Total time of calculations for extraction of candidates for all three models, using naive procedure was 10 ms, and using optimized procedure it was 10 ms.

7.2.3 Asymmetric Weighted Hamming distance

Category	$\lvert Candidates\rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core\rvert$	$\lvert Outer\rvert$	$\lvert Boundary\rvert$
lung	1	none	2.30	4.52	11	19	21
stomach	1	none	0.61	3.67	2	3	46
ovary	1	none	0	1.39	1	29	21

Table 7.11: Tumor dataset, Asymmetric Weighted Hamming distance, centroids.

Centroid version The table 7.11 shows for each of the categories the size of the set of candidates returned by the *extractCandidates* function and the prototype selected for the model. As one can see, none of the candidates for all three categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned. For this reason, the values of the radii and sizes of individual model areas presented in the table refer to the prototype candidate. One can notice a relatively small size of the cores and large boundaries, which suggests the low quality of the obtained models. Overall, the quality of the results appears to be better than for unweighted Hamming distance, but worse than for symmetrical Weighted Hamming distance (although the size of the core of *stomach* category is bigger in case of AWH).

Since all three categories are ill-defined, the agent does not generate any statements about membership of the objects to the above categories.

Total time of calculations for extraction of candidates for all three models, using naive procedure was 14058 ms, and using optimized procedure it was 27 ms.

Category	$\|Candidates\|$	**Prototype**	τ_c^+	τ_c^-	$\|Core\|$	$\|Outer\|$	$\|Boundary\|$
lung	1	none	1.47	4.86	5	4	42
stomach	1	none	0	3.32	1	7	43
ovary	1	none	0	1.39	1	29	21

Table 7.12: Tumor dataset, Asymmetric Weighted Hamming distance, medoids.

Medoid version The results are identical to the centroid version for *ovary* category. However, for *lung* and *stomach* categories the sizes of the boundaries have been bigger in case of medoids. As one can see, none of the candidates for all three categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned.

Since all three categories are ill-defined, the agent does not generate any statements about membership of the objects to the above categories.

Total time of calculations for extraction of candidates for all three models, using naive procedure was 24 ms, and using optimized procedure it was 16 ms.

7.3 1984 United States Congressional Voting Records Database

The data set „1984 United States Congressional Voting Records Database” was downloaded from https://archive.ics.uci.edu/ml/datasets/Congressional+Voting+Records.

The data set contains 435 examples (objects) representing members of U.S. Congress, divided into two classes (categories) depending on their party affiliation:

- *democrat* – 267 examples,
- *republican* – 168 examples.

This data set includes votes for each of the U.S. House of Representatives Congressmen on the 16 key votes identified by the Congressional Quarterly Almanac[10]:

- handicapped-infants: nay(0), yea(1), unknown(2),
- water-project-cost-sharing: nay(0), yea(1), unknown(2),
- adoption-of-the-budget-resolution: nay(0), yea(1), unknown(2),
- physician-fee-freeze: nay(0), yea(1), unknown(2),
- el-salvador-aid: nay(0), yea(1), unknown(2),
- religious-groups-in-schools: nay(0), yea(1), unknown(2),
- anti-satellite-test-ban: nay(0), yea(1), unknown(2),
- aid-to-nicaraguan-contras: nay(0), yea(1), unknown(2),
- mx-missile: nay(0), yea(1), unknown(2),
- immigration: nay(0), yea(1), unknown(2),
- synfuels-corporation-cutback: nay(0), yea(1), unknown(2),
- education-spending: nay(0), yea(1), unknown(2),
- superfund-right-to-sue: nay(0), yea(1), unknown(2),
- crime: nay(0), yea(1), unknown(2),
- duty-free-exports: nay(0), yea(1), unknown(2),
- export-administration-act-south-africa: nay(0), yea(1), unknown(2).

Actually, the CQA lists nine different types of votes:

- voted for, paired for, and announced for – these three simplified to "yea",
- voted against, paired against, and announced against – these three simplified to "nay",
- voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known – these three simplified to an "unknown" disposition.

The size of the universe of mental representations of distinguishable objects O is 3^{16}, which is over 43 millions of objects. It is large number, so it was impossible to perform in reasonable time (less than 24 hours) calculations for naive algorithm in case of centroids. The calculation times for more efficient algorithms will be shown in the following chapters. Simulations were conducted on computer with Intel(R) Core(TM) i5-4310U CPU @ 2.00-2.60 GHz and 8GB RAM. The algorithms were implemented using C#/.NET framework. Let us note, that calculation times are not very accurate, since they were measured in multitasking operation system (MS Windows 10) and not in real time operating system. Also, .NET background processes like garbage collector can affect the performance. Still, the times are accurate enough to show differences between implemented algorithms.

7.3.1 Hamming distance

Category	$\lvert Candidates \rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
democrat	1	none	3	16	57	7	278
republican	1	none	2	NULL	35	0	307

Table 7.13: Voting dataset, Hamming distance, centroids.

Centroid version The table 7.13 shows for each of the categories the size of the set of candidates returned by the *extractCandidates* function and the prototype selected for the model. As one can see, none of the candidates for both categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned. For this reason, the values of the radii and sizes of individual model areas presented in the table refer to the prototype candidate. One can notice a relatively small size of the cores and large boundaries, which suggests the low quality of the obtained models.

Since both categories are ill-defined, the agent does not generate any statements about membership of the objects to the above categories.

Total time of calculations for extraction of candidates for both models, using optimized procedure was 16 ms.

Medoid version The results are shown in table 7.14, they are identical to the case with centroids. Again, none of the candidates for both categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned.

Total time of calculations for extraction of candidates for both models, using naive procedure was 47 ms, and using optimized procedure it was 17 ms.

Category	$\lvert Candidates \rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
democrat	1	none	3	16	57	7	278
republican	1	none	2	NULL	35	0	307

Table 7.14: Voting dataset, Hamming distance, medoids.

7.3.2 Weighted Hamming distance

Category	$\lvert Candidates \rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
democrat	3	[11100011 10100011]	1.09	3.98	136	41	165
republican	3	none	0.21	NULL	29	0	313

Table 7.15: Voting dataset, Weighted Hamming distance, centroids.

Centroid version The table 7.15 shows for each of the categories the set of candidates returned by the *extractCandidates* function and the prototype selected for the model. As can be seen, prototype has been found for one of two categories. The values of the radii and the size of the individual areas of the model presented in the table refer to the prototype, or to the candidate if category is ill-defined.

The prototypical congressman for category *democrat* has the following characteristics: handicapped-infants=yea, water-project-cost-sharing=yea, adoption-of-the-budget-resolution=yea, physician-fee-freeze=nay, el-salvador-aid=nay, religious-groups-in-schools=nay, anti-satellite-test-ban=yea, aid-to-nicaraguan-contras=yea, mx-missile =yea, immigration=nay, synfuels-corporation-cutback=yea, education-spending=nay, superfund-right-to-sue=nay, crime=nay, duty-free-exports=yea, export-administration-act-south-africa=yea.

For the *democrat* category model, a prototype was found, but for the *republican* category model, none of the candidates has met the qualitative condition. One can also see, that the core of the *democrat* model contains four times more objects than for *republican*. This could lead to the conclusion, that the Democrats constitute a more homogeneous environment than the Republicans, for whom it is impossible to find a prototype congressman.

It is also interesting to compare the voting of the (proto)typical member of each party. Since for Republicans it has been impossible to find the prototype satisfying quality criteria, one of the candidates will be used for comparison:

- *democrat* prototype [1110001110100011],
- *republican* candidate [0201110001011101].

Looking at the values of the attributes for both categories representatives, it can be seen that in almost all cases a typical Republican votes differently than a typical Democrat. The exception is the last attribute (export-administration-act-south-africa) for which the values are the same. Certainly, the above conclusions should be verified by a political science expert, but the resulting models of categories are interesting starting points for more elaborate analysis, like for example, how close the particular congressmen are to the prototypical one.

Total time of calculations for extraction of candidates for both models using optimized procedure was 48 ms.

Cognitive structures created in the learning process on the basis of the above data can be used by the agent to generate statements in accordance with the assumed cognitive semantics for category *democrat*. Examples of valid grounded formulas are presented below:

- for the object $x = [1110001110000022]$ the epistemic satisfaction relation holds for formula $Know(x \in democrat)$, because the corresponding object o is at the distance of 0.43 from the prototype, so in the core of model of category *democrat*,
- for the object $x = [0101110000011102]$ the epistemic satisfaction relation holds for formula $Know(x \notin democrat)$, because the corresponding object o is at the distance of 4.01 from the prototype, so in the outer region of model of category *democrat*,
- for the object $x = [0101110000002111]$ the corresponding object o lies at the distance of 3.39 from the prototype, i.e. within the boundary of model of category *democrat*; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{democrat} = 1.448$; for this size of the epistemic neighborhood $EN_{democrat}(o, \varepsilon_{democrat})$ of the object o, the relative grounding strength is $\lambda_{democrat}(EN_{democrat}(o, \varepsilon_{democrat})) = 0.13$; hence for $\lambda_{minBel} = 0.5$ the epistemic satisfaction relation holds for formulas $Bel(x \notin democrat)$ and $Pos(x \in democrat)$,
- for the object $x = [1110110000121111]$ the corresponding object o lies at the distance of 2.36 from the prototype, i.e. within the boundary of model of category *democrat*; assuming the coefficient of radius of epistemic neighborhood $\alpha = 0.5$ we get the radius of epistemic neighborhood $\varepsilon_{democrat} = 1.448$; for this size of the epistemic neighborhood $EN_{democrat}(o, \varepsilon_{democrat})$ of the object, the relative grounding strength is $\lambda_{democrat}(EN_{democrat}(o, \varepsilon_{democrat})) = 0.94$; hence for $\lambda_{minBel} = 0.5$ the epistemic satisfaction relation holds for formulas $Bel(x \in democrat)$ and $Pos(x \notin democrat)$.

Category	$\lvert Candidates \rvert$	**Prototype**	τ_c^+	τ_c^-	$\lvert Core \rvert$	$\lvert Outer \rvert$	$\lvert Boundary \rvert$
democrat	3	[11100011 10100011]	1.09	3.98	136	41	165
republican	3	none	0.21	NULL	29	0	313

Table 7.16: Voting dataset, Weighted Hamming distance, medoids.

Medoid version The results are shown in table 7.16, they are identical to the case with centroids.

Total time of calculations for extraction of candidates for both models, using naive procedure was 79 ms, and using optimized procedure it was 49 ms.

7.3.3 Asymmetric Weighted Hamming distance

Category	$\|Candidates\|$	**Prototype**	τ_c^+	τ_c^-	$\|Core\|$	$\|Outer\|$	$\|Boundary\|$
democrat	1	none	3.54	12.83	90	29	223
republican	1	none	1.33	NULL	27	0	315

Table 7.17: Voting dataset, Asymmetric Weighted Hamming distance, centroids.

Centroid version The table 7.17 shows for each of the categories the size of the set of candidates returned by the *extractCandidates* function and the prototype selected for the model. As one can see, none of the candidates for both categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned. For this reason, the values of the radii and sizes of individual model areas presented in the table refer to the prototype candidate. One can notice a relatively small size of the cores and large boundaries, which suggests the low quality of the obtained models.

Since both categories are ill-defined, the agent does not generate any statements about membership of the objects to the above categories.

Total time of calculations for extraction of candidates for both models, using optimized procedure was 104 ms.

Medoid version The results are shown in table 7.18, they are identical to the case with centroids. Again, none of the candidates for both categories has met the qualitative condition, therefore no prototype has been selected and no category has been learned.

Total time of calculations for extraction of candidates for both models, using naive procedure was 121 ms, and using optimized procedure it was 61 ms.

Category	$\|Candidates\|$	**Prototype**	τ_c^+	τ_c^-	$\|Core\|$	$\|Outer\|$	$\|Boundary\|$
democrat	1	none	3.54	12.83	90	29	223
republican	1	none	1.33	NULL	27	0	315

Table 7.18: Voting dataset, Asymmetric Weighted Hamming distance, medoids.

7.4 Conclusions

The simulations confirmed the results of theoretical analyzes of the computational complexity of the algorithms for candidate extraction presented in chapter 6. By far the most computationally expensive was the naive centroid search procedure. As anticipated, the above procedure has proved to be impractical for big data, due to exponential complexity, resulting in very long time of calculations. The differences between the other algorithms were not large, but the optimized procedures still ran faster, both for centroids and medoids extraction.

Limiting the set of potential candidates to medoids allowed for a slightly shorter computation time for some data sets than for centroids. However, for the "Primary Tumor Domain" data set and the Weighted Hamming model, no medoid candidate has met the qualitative condition and hence a medoid prototype could not be found, whereas for the centroid case it was possible. Therefore, it seems that it is not worth limiting the set of potential candidates to medoids, unless it is required by the problem domain, that is, when it is necessary for the prototype to represent the object actually experienced by the agent.

When it comes to comparing the quality of models, in this work it is considered that the smaller the model boundary, the higher its quality. Of course, well-defined models of category (with designated prototype) are also better than ill-defined models (without prototype). From that point of view, the case of Hamming distance without weights is by far the worst among the macrostructures used in implementation models. No prototype could be found for any of the models (centroid or medoid) for any of the actual data sets. The size of boundaries for all data sets were also much bigger than for weighted Hamming distances.

The models using symmetrical Weighted Hamming distance turned out to be of better quality than the models using Asymmetrical Weighted Hamming distance. The former usually have had smaller boundary and also they have been able to designate prototypes for two categories (*lung* and *democrat*) for which this has not been possible for the latter ones. Therefore, less complex (symmetrical) model is better for examined data sets. However, the quality differences were not very large, so it would probably be worth exploring both macrostructures with more data sets.

Overall, the well-defined models (with designated prototypes) have been found for most of the studied categories:

- categories *none*, $soft$ and *hard* from "Database for fitting lenses" data set,
- category *lung* from "Primary Tumor Domain" data set,
- category *democrat* from "1984 United States Congressional Voting Records Database" data set.

For the following categories, the models were ill-defined (candidates have not met quality conditions):

- categories *stomach* and *ovary* from "Primary Tumor Domain" data set,

- category *republican* from "1984 United States Congressional Voting Records Database" data set.

The mere fact that we can identify which models are of poor quality is already an advantage. Now it is possible to try to find a better model, e.g. by removing any outliers (possibly containing corrupted data) or using a different macrostructure. It is also possible, that the model presented in this work simply does not fit the structure of the category, which is socially sanctioned and/or imposed by physical factors. But it can also be an interesting direction for analysis by domain experts, that could answer the questions like "why is it possible to designate (proto)typical Democrat while it is hard to designate (proto)typical Republican?" or "why is it possible to designate (proto)typical patient with lung tumor while it is hard to designate (proto)typical patient with stomach tumor?".

As it was mentioned, the models of categories can be a basis for further analysis conducted by domain experts. For example, the issues studied may include:

- the features of the prototype of category, e.g. prototypical patient with lung cancer,
- the features common to elements in the core of the model, e.g. members of the party,
- the differences between the prototypes of the (allegedly) opposing categories, e.g. the Democrats and the Republicans,
- identification of outliers, e.g. atypical cases of tumors.

Chapter 8

Summary

The aim of this dissertation was **development and verification** of an artificial agent system capable of generating statements about the membership of an object to a category, the behavior of which in this respect would be consistent with human linguistic practice.

The realization of the above goal was preceded by an analysis of literature in the field of cognitive psychology and linguistics, discussing the results of research on the structure and practical role of categories with a prototype in natural language processing (e.g. [38, 72, 74]) and the so-called theory of grounding of the modal language of communication in agent systems, which takes into account the case of a model of generating statements about the membership of an object to the category *without the prototype* (e.g. [31, 33]).

The analysis of the state of knowledge showed the lack of satisfactory studies on generating modal statements about the object's membership to the category with the prototype. Therefore, it was assumed that from the theoretical and practical point of view, it is desirable to equip the artificial agent with its own embodied ontology, including models of categories with a prototype, and to develop a computationally realizable definition of cognitive semantics describing the sense of statements about the object's membership to this group of conceptual categories.

The author in the dissertation proposes an original supplement to the existing theoretical and technical solutions. The main achievements complementing the above solutions concern the realization of the following major objectives:

(1) development of a general strategy for learning the category's model with a prototype and analyzing the properties of this strategy,

(2) development of a technically oriented definition of cognitive semantics for a class of modal statements about the object's membership to the category with a prototype, for the cases of objects with complete and incomplete information

about their state,

(3) demonstration of the compliance of the developed cognitive semantics with the pragmatics of the use of modal statements by humans about the object's membership to the category with a prototype,

(4) development and analysis of the properties of three original implementation models of the proposed strategy of learning the model of category with a prototype.

In order to achieve the above goals, the author proposed solutions to the related problems and performed a number of detailed tasks, presented below.

The achievement of the first objective included:

- defining the internal representation of the objects of the external world and how they are processed in the agent's mental space (def. 4.1 – 4.3),
- defining a model of category with a prototype, taking into account the division into the core, the boundary and the outer region of the category (def. 4.4),
- developing a general strategy for learning the model of the category with a prototype (chapter 4.4.1),
- formulation of the condition determining the correctness of the category's model (algorithm 4.2, line 26),
- the proof of theorem 4.1 showing that the model obtained as a result of applying the developed strategy maintains the correct relationship between the core, the boundary and the outer region of the learned category with a prototype,
- the proof of theorem 4.2 on the computational complexity of the developed strategy of learning the category with a prototype,
- demonstrating that the proposed model of category with a prototype and the strategy for its learning meet the six theses of the standard version of the prototype semantics developed by Rosch (chapter 4.5).

The achievement of the second objective included:

- defining the syntax of the modal categorization language, including modal statements about the object's membership to a category, using the modal operators of knowledge, belief and possibility (def. 5.1 – def. 5.3),
- defining the concept of epistemic neighborhood (def. 5.7 and 5.16),

- adaptation of the concept of relative grounding strength of modal statements, used in so-called the theory of grounding of modal communication language in agent systems, for the purposes of cognitive semantics proposed in the dissertation (def. 5.8),
- the detailed discussion of the proposed method of applying the cognitive semantics of modal statements about the object's membership to the category with a prototype for the case of objects with *complete* representation of their state (algorithm 5.1),
- for objects with *complete* representation of their state, developing a set of so-called epistemic satisfaction relations of modal formulas representing individual cases of cognitive semantics of modal statements about membership of the object to the category with a prototype (def. 5.5, 5.6, 5.9, 5.10),
- the detailed discussion of the proposed method of applying the cognitive semantics of modal statements about the object's membership to the category with a prototype for the case of objects with *incomplete* representation of their state (algorithm 5.3),
- for objects with *incomplete* representation of their state, developing a set of so-called epistemic satisfaction relations of modal formulas representing individual cases of cognitive semantics of modal statements about membership of the object to the category with a prototype (def. 5.14, 5.15, 5.17, 5.18).

The achievement of the third objective included:

- the proof of theorems 5.1 – 5.12 on the compliance of the developed cognitive semantics with the pragmatics of the use of modal statements by humans about the object's membership to the category with a prototype,
- discussion of the relationship between the proposed cognitive semantics and the theory of modal grounding of the language of communication in agent systems, formulated for the case of categories without a prototype (chapter 5.6).

The achievement of the fourth objective included:

- for the implementation model based on the Hamming distance in the version using the centroid selection mechanism (chapters 6.2.2 – 6.2.3):
  - analysis of exponential computational complexity (theorem 6.1) of naive search for candidates,
  - a proposal of an optimized algorithm (6.1) along with a proof of its polynomial computational complexity for practical applications (theorem 6.8),

- for the implementation model based on the Hamming distance in the version using the medoid selection mechanism (chapters 6.2.4 – 6.2.5):
  - analysis of polynomial computational complexity (theorem 6.9) of naive search for candidates,
  - a proposal of an optimized algorithm (6.2) along with a proof of its lower (albeit polynomial) computational complexity (theorem 6.13),
- for the implementation model based on the Weighted Hamming distance in the version using the centroid selection mechanism (chapters 6.3.2 – 6.3.3):
  - analysis of exponential computational complexity (theorem 6.14) of naive search for candidates,
  - a proposal of an optimized algorithm (6.3) along with a proof of its polynomial computational complexity for practical applications (theorem 6.21),
- for the implementation model based on the Weighted Hamming distance in the version using the medoid selection mechanism (chapters 6.3.4 – 6.3.5):
  - analysis of polynomial computational complexity (theorem 6.22) of naive search for candidates,
  - a proposal of an optimized algorithm (6.4) along with a proof of its lower (albeit polynomial) computational complexity (theorem 6.24),
- for the implementation model based on the Asymmetric Weighted Hamming distance in the version using the centroid selection mechanism (chapters 6.4.2 – 6.4.3):
  - analysis of exponential computational complexity (theorem 6.25) of naive search for candidates,
  - a proposal of an optimized algorithm (6.5) along with a proof of its polynomial computational complexity for practical applications (theorem 6.31),
- for the implementation model based on the Asymmetric Weighted Hamming distance in the version using the medoid selection mechanism (chapters 6.4.4 – 6.4.5):
  - analysis of polynomial computational complexity (theorem 6.32) of naive search for candidates,
  - a proposal of an optimized algorithm (6.6) along with a proof of its lower (albeit polynomial) computational complexity (theorem 6.36),
- implementation (chapter 7) of the above models and strategies in a computer system, which ultimately proves their technical feasibility,
- conducting simulation tests (chapter 7) using the above implementation on machine learning datasets downloaded from the Machine Learning Repository provided

by the University of California, Irvine:

- "Database for fitting contact lenses",
- "1984 United States Congressional Voting Records Database",
- "Primary Tumor Domain",

- empirical confirmation of the compliance of the theoretically determined computational complexity of the developed mechanisms with the calculation time needed to process real data sets (e.g. chapters 7.2.1 – 7.2.3),
- presentation of conclusions from the results of the simulations (chapter 7.4), among others:
  - empirical confirmation of theoretical predictions regarding better applicability of implementation models using as macrostructures distance functions with weights,
  - examples of using the learned model of a category as a starting point for domain analysis, e.g. identifying prototypical representatives of certain categories, such as a typical Democratic congressman or a person with lung cancer.

The strategy, cognitive semantics and implementation models analyzed in this dissertation can be used as a basis for further research in the areas presented below.

1. Investigation of more implementation models that use different macrostructures (e.g. similarity measures instead of distance) and mechanisms for determining the set of candidates for the prototype. An interesting review of similarity measures together with postulates regarding their properties, similar to those presented in this dissertation, can be found, for example, in [48].

2. Examination of models based on microstructures other than enumerated attributes, e.g. attributes with a continuous set of values, trees or graphs. This involves the development of appropriate macrostructures. Examples of distance, similarity and dissimilarity measures for graphs can be found in [68, 70, 96].

3. Extending the model of category to use more than one prototype. This would make it possible to model non-radial categories, e.g. categories based on family resemblance.

4. Developing a strategy for the simultaneous use of several models for the same category, which would allow one to find the most useful model. The use of a particular model could depend on the context in which the category is used. Using the right context is necessary to obtain statements that are relevant to real situations[76].

5. Modifying the category model by allowing the core to contain a certain number (e.g. a percentage) of negative examples and/or the outer region to contain a certain

number of positive examples. Such a model would probably have greater resistance to noise and faulty attribute values.

6. Extending the embodied ontology with relations between categories, such as synonymy, antonymy, subsumption, etc. Relationship definitions would probably have to refer to the internal structure of the category model, i.e. the prototype and the three areas of the model.

7. Implementation of the embodied agent (e.g. as a robot) and its confrontation with the physical environment and the expectations of a human communication partner.

List of tables

List of figures

References

[1] Amini A., Schwarting W., Soleimany A. Rus D.: *Deep Evidential Regression*, in: Advances in Neural Information Processing Systems, Vol. 33, 14927-14937, Curran Associates, Inc., 2020.

[2] Amodei D., Hernandez D., OpenAI: *AI and Compute*, https://openai.com/blog/ai-and-compute/, 2018.

[3] Babuska R.: *Construction of fuzzy systems-interplay between precision and transparency*, in: ESIT 2000 European Symposium on Intelligent Techniques, Aachen - S.l., 445-452, ERUDIT, 2001.

[4] Beach L.R.: *Cue probabilism and inference behavior*, in: Psychological Monographs, 78 (Whole No. 582), 1964.

[5] Beach L.R.: *Recognition, assimilation, and identification of objects*, in: Psychological Monographs, 78 (Whole No. 583), 1964.

[6] Bender E.M., Gebru T., McMillan-Major A., Shmitchell S.: *On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?*, in: FAccT '21: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610–623, 2021.

[7] Bermudez J.L.: *Cognitive Science: An Introduction to the Science of the Mind*, Cambridge University Press, 2014.

[8] Brockman G., Murati M., Welinder P., OpenAI: *OpenAI API*, https://openai.com/blog/openai-api/, 2020.

[9] Brown T.B. et al.: *Language Models are Few-Shot Learners*, arXiv:2005.14165, 2020.

[10] Congressional Quarterly Almanac, 98th Congress, 2nd session 1984, Volume XL: Congressional Quarterly Inc., Washington, D.C., 1985.

[11] Davenport T., Ronanki R.: *The rise of cognitive agents. Will humans prefer to talk with machines?*, Deloitte University Press, 2016.

[12] Diefenbach S., Gerber N., Hassenzahl M.: *The 'Hedonic' in Human-Computer Interaction – History, Contributions, and Future Research Directions*, DIS 2014 Conference on Designing Interactive Systems, 2014.

[13] Estivill-Castro V.: *Why so many clustering algorithms: a position paper*, ACM SIGKDD Explorations Newsletter vol. 4(1)/2002, 65–75, 2002.

[14] Fedus W., Zoph B., Shazeer N.: *Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity*, arXiv:2101.03961, 2021.

[15] Gärdenfors P.: *Conceptual Spaces: The Geometry of Thought*, A Bradford Book, MIT Press, Cambridge Massachusetts, 2000.

[16] Gärdenfors P., Williams M.A.: *Reasoning about Categories in Conceptual Spaces.*, Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI 2001, Seattle, Washington, USA, August 4-10, 2001.

[17] Gärdenfors P.: *Conceptual Spaces as a Framework for Knowledge Representation*, Mind and Matter, 2, 9-27, 2004.

[18] Geeraerts D.: *On Necessary And Sufficient Conditions*, Journal of Semantics, Volume 5, Issue 4, 275–291, 1986.

[19] Geeraerts D.: *Prospects and problems of prototype theory*, in: Diacronia, 8, 2016.

[20] Geeraerts D., Cuyckens H.: *The Oxford Handbook of Cognitive Linguistics*, Oxford University Press, 2007.

[21] Greco S., Matarazzo B., Słowiński R., *The use of rough sets and fuzzy sets in MCDM*, In: Gal T., Stewart T., Hanne T. (Eds) Advances in Multiple Criteria Decision Making, Kluwer, chapter 14, 14.1-14.59, 1999.

[22] Grzegorczykowa R., *Wprowadzenie do semantyki językoznawczej*, Wyd. Naukowe PWN, Warszawa, 2010.

[23] Harnad S.: *The Symbol Grounding Problem*, in: Physica D 42, 335-346, 1990.

[24] Harnad S.: *To Cognize is to Categorize: Cognition is Categorization*, in: Handbook of Categorization in Cognitive Science (Second Edition), 21-54, Elsevier, 2017.

[25] Hartigan J.A.: *Clustering Algorithms.* New York, Wiley, 1975.

[26] Hartigan J.A., Wong M.A.: *Algorithm AS 136: A K-Means Clustering Algorithm*, in: Journal of the Royal Statistical Society. Series C (Applied Statistics), Vol. 28, No. 1(1979), 100-108, 1979.

[27] Heinert M.: *Artificial neural networks–how to open the black boxes*, in: Application of Artificial Intelligence in Engineering Geodesy, Reiterer, A. and Egly, U. (Eds.), Proceedings: 42–62, Vienna, Austria, 2009.

[28] Hindle D.: *Noun Classification From Predicate-Argument Structures*, in: 28th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Pittsburgh, Pennsylvania, USA, 268-275, 1990.

[29] Janssen C.P., Donker S.F., Brumby D.P., Kun A.L.: *History and future of human-automation interaction*, in: International Journal of Human-Computer Studies, vol. 131, 99-107, 2019.

[30] Kanungo T., Mount D.M., Netanyahu N.S., Piatko C.D., Silverman R., Wu A.Y.: *An efficient k-means clustering algorithm: analysis and implementation*, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, 881-892, July 2002.

[31] Katarzyniak R.: *On some properties of grounding simple modalities.*, in: Systems Science, vol. 31(3), 59-86, 2005.

[32] Katarzyniak R.: *Gruntowanie modalnego języka komunikacji w systemach agentowych*, Wyd. Exit, Warszawa 2007.

[33] Katarzyniak R., Mulka M., Popek G., Żurawski M.: *Towards Communicative Agents with Cognitive Semantics of Modal Class-Membership Statements*, in: 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, Changsha, China, 1981-1986, 2016.

[34] Katarzyniak R., Lorkiewicz W., Mulka M., Krejcar O.: *Towards basic level categories in cognitive agents*, in: 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, Changsha, China, 400-407, 2017.

[35] Kaufman L., Rousseeuw P.: *Clustering by Means of Medoids*, in: Data Analysis based on the L1-Norm and Related Methods, 405-416, 1, 1987.

[36] Kay P., McDaniel C.: *The Linguistic Significance of the Meanings of Basic Color Terms*, in: Language, Vol. 54, No. 3 (Sep., 1978), pp. 610-646, 1978.

[37] Kendall A., Gal Y.: *What uncertainties do we need in bayesian deep learning forcomputer vision?*, in: Advances in neural information processing systems, 5574–5584, 2017.

[38] Kleiber G.: *La semantique du prototype. Categories et sens lexical*, Presses Universitaires de France, 1990.

[39] Kumar V., Boddeti V.N., Smereka J.M., Thornton J., Savvides M.: *Chapter 15 - Application of Bayesian Graphical Models to Iris Recognition*, in: Handbook of Statistics, vol. 31, 381-398, Elsevier, 2013.

[40] Laird J.E.: *The Soar Cognitive Architecture*, The MIT Press, 2012.

[41] Lakoff G.: *Categories*, in: Linguistics in the Morning Calm, Linguistic Society of Korea, eds., Seoul: Hanshin, 1982.

[42] Lakoff G.: *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*, University of Chicago Press, 1987.

[43] Langacker R.W.: *Foundations of Cognitive Grammar*, vol.I, Stanford, Stanford University Press, 1987.

[44] Langacker R.W.: *Cognitive Grammar: A Basic Introduction*, Oxford, Oxford University Press, 2008.

[45] Lechniak M.: *Czym są modalności i jak je rozróżniać? Próby klasyfikacji modalności*, In: Jedność i wielość logik modalnych, red. Tkaczyk M., 63-80, Lublin, 2019.

[46] Lee J.H., Kim M.H., Lee Y.J.: *Information retrieval based on conceptual distance in is-a hierarchies*, In: Journal of Documentation, vol.49, no.2, 188-207, 1993.

[47] Lesot M.-J., Rifqi M., Benhadda H.: *Similarity measures for binary and numerical data: a survey*, In: International Journal of Knowledge Engineering and Soft Data Paradigms (IJKESDP), Vol. 1, No. 1, 63-84, 2009.

[48] Lin D.: *An Information-Theoretic Definition of Similarity*, In: Proceedings of the 15th International Conference on Machine Learning, 296-304, Morgan Kaufmann, 1998.

[49] Loftus E.: *Category dominance, instance dominance, and categorization time*, in: Journal of Experimental Psychology, 97, 70-74, 1973.

[50] Malinin A., Gales M.: *Predictive uncertainty estimation via prior networks*, in: Advances in Neural Information Processing Systems, 7047–7058, 2018.

[51] McGill M.: *An Evaluation of Factors Affecting Document Ranking by Information Retrieval Systems*, report, Syracuse Univ., NY, School of Information Studies, 1979.

[52] Mervis C.B., Pani J.R.: *Acquisition of basic object categories*, in: Cognitive Psychology, 12(4), 496-522, 1980.

[53] Miller G.A., *The cognitive revolution: a historical perspective*, in: Trends in Cognitive Sciences, 7(3), 141-144, 2003.

[54] Mitchell T.M.: *Machine Learning*, McGraw-Hill Science/Engineering/Math, 1997.

[55] Nęcka E., Orzechowski J., Szymura B.: *Psychologia poznawcza*, Wydawnictwo Naukowe PWN, Warszawa 2006.

[56] Nilsson J., Sokoler T., Binder T., Wetcke N.: *Beyond the Control Room: Mobile Devices for Spatially Distributed Interaction on Industrial Process Plants*, In: Lecture Notes in Computer Science 1927, 30-45, 2000.

[57] National Transportation Safety Board: *Supervisory Control and Data Acquisition (SCADA) in Liquid Pipelines*, Safety Study NTSB/SS-05/02, Washington, DC, 2006.

[58] Nowak M.A., Krakauer D.C.: *The evolution of language*, Proceedings of the National Academy of Sciences, 96 (14), 8028-8033, Jul 1999.

[59] Núñez R., Allen M., Gao R. et al.: *What happened to cognitive science?*, Nature Human Behaviour, 3, 782–791, 2019.

[60] Ogden C., Richards I.: *The meaning of meaning*, Harcourt, Brace, 1923.

[61] Ochsner K., Kosslyn S.M.: *The Oxford Handbook of Cognitive Neuroscience*, Oxford University Press, 2013.

[62] Palmer F.R.: *Mood and Modality*, 2nd ed., Cambridge: Cambridge University Press, 2001.

[63] Pawlak Z.: *Rough sets*, in: International Journal of Computer and Information Sciences 11, s.341–356, 1982.

[64] Pawlak Z.: *Systemy informacyjne. Podstawy teoretyczne*, WNT Warszawa, 1983.

[65] Pelleg D., Moore A.: *Accelerating exact k-means algorithms with geometric reasoning*, in: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '99), Association for Computing Machinery, New York, NY, USA, 277–281, 1999.

[66] Perzanowski J.:*Ontyka modalności*, In: Myśli o języku, nauce, wartościach, W. Strawiński, M. Grygianiec, A. Brożek, Warszawa, Wyd. Naukowe Semper, 281-297, 2006.

[67] Petrick E.R.:*A Historiography of Human–Computer Interaction*, In: IEEE Annals of the History of Computing, vol. 42(4), 8-23, 2020.

[68] Rada R.„ Mili H., Bicknell E., Blettner M.: *Development and application of a metric on semantic nets*, In: IEEE Transactions on Systems, Man, and Cybernetics, 19, 17-30, 1989.

[69] Reed S.K.: *Pattern recognition and categorization*, Cognitive Psychology, 3, 382-407, 1972.

[70] Resnik P.: *Using Information Content to Evaluate Semantic Similarity in a Taxonomy*, Proceedings of the 14th international joint conference on Artificial intelligence - Volume 1 , page 448-453. San Francisco, CA, USA, Morgan Kaufmann Publishers Inc., 1995.

[71] Ritter F. E., Tehranchi F., Oury J. D.: *ACT-R: A cognitive architecture for modeling cognition*, Wiley interdisciplinary reviews. Cognitive science, 10(3), 2019.

[72] Rosch, E.: *Natural Categories*, in: Cognitive Psychology 4, 328-350, 1973.

[73] Rosch, E.: *Cognitive Reference Points*, in: Cognitive Psychology 7, 532-547, 1975.

[74] Rosch, E., Mervis, C.: *Family Resemblances: Studies in the Internal Structure of Categories*, in: Cognitive Psychology 7, s.573-605, 1975.

[75] Rosch, E. et al.: *Basic Objects in Natural Categories*, in: Cognitive Psychology 8, 382-436, 1976.

[76] Rosch, E.: *Principles of Categorization*, in: Foundations of cognitive psychology : core readings, ed. by Daniel J. Levitin., 251-270, Cambridge Mass., London England, MIT Press, 2002.

[77] Roy D.: *Semiotic schemas: A framework for grounding language in action and perception*, Artificial Intelligence, vol. 167(1-2), 170–205, 2005.

[78] Rudin C.: *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, Nature Machine Intelligence, 1, 206–215, 2019.

[79] Rudin C., Radin J.: *Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson From an Explainable AI Competition*, Harvard Data Science Review, 1(2), 2019.

[80] Russell S., Norvig P.: *Artificial Intelligence: A Modern Approach (3rd ed.)*, Prentice Hall Press, Upper Saddle River, NJ, USA, 2009.

[81] Samsonovich A.: *Toward a Unified Catalog of Implemented Cognitive Architectures*, in: Biologically Inspired Cognitive Architectures 2010, IOS Press, Amsterdam, 195-244, 2010.

[82] Schubert E., Rousseeuw P.J.: *Fast and eager k-medoids clustering: O(k) runtime improvement of the PAM, CLARA, and CLARANS algorithms*, in: Information Systems, vol. 101, 2021.

[83] *Cognitive Science, 1978: Report of The State of the Art Committee to The Advisors of The Alfred P. Sloan Foundation*, 1978.

[84] Steels L.: *Grounding Language through Evolutionary Language Games*, In: Language Grounding in Robots, 1-22, Steels L., Hild M.(eds), Springer, 2012.

[85] Stefanowski J.: *On rough set based approaches to induction of decision rules*, In: Rough sets in knowledge discovery 1, 500-529, 1999.

[86] Stefanowski J., Żurawski M.:*Incremental Rule Induction for Multicriteria and Multiattribute Classification*, In: Intelligent Information Processing and Web Mining Proceedings of the International IIS: IIPWM´03 Conference held in Zakopane, Poland, June 2-5, 2003, Editors: Klopotek M. A., Wierzchon S. T., Trojanowski K. Series: Advances in Soft Computing. Springer, 2003.

[87] Tversky A.: *Features of similarity*, In: Psychological Review, 84(4), 327–352, 1977.

[88] Urbaniak A.: *Podstawy automatyki*, Wydawnictwo Politechniki Poznańskiej, 2007.

[89] Vogt P.:*The physical symbol grounding problem*, in: Cognitive Systems Research Journal, vol. 3, 429-457, 2002.

[90] Vogt P.:*Language evolution and robotics: Issues on symbol grounding and language acquisition*, in: Artificial Cognition Systems, Hershey, PA.: Idea Group, 176-209, 2006.

[91] Wang G., Peng M., Wang H., Zhang Y., Xie Q., Tian G.: *Incorporating word embeddings into topic modeling of short text.* In: Knowledge and Information Systems 61, 1123–1145, 2019.

[92] Więcek D.: *Grounding of modal equivalences in agent systems* (in Polish), PhD thesis, 2019.

[93] Wilkins A.: *Conjoint frequency, category size, and categorization time*, in: Journal of Verbal Learning and Verbal Behavior, 10, 382-385, 1971.

[94] Wilson R.A., Keil F.C.: *The MIT encyclopedia of the cognitive sciences*, Cambridge, Mass. : MIT Press, 1999.

[95] Wittgenstein L.: *Philosophical Investigations*, Wiley-Blackwell, 1953.

[96] Wu H., Su Z., Mao F., Olman V., Xu Y.: *Prediction of functional modules based on comparative genome analysis and Gene Ontology application*, in: Nucleic Acids Research, 33(9), 2822-2837, 2005.

[97] Wu Z.F., Li J., Cai M.Y., Lin Y., Zhang W.J.: *On membership of black-box or white-box of artificial neural network models*, in: 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), , 1400-1404, 2016.

[98] Zadeh L.: *Fuzzy Sets*, in: Information and Control 8, 338-353, 1965.

[99] Zeigenfuse M., Lee M.: *A Comparison of Three Measures of the Association Between a Feature and a Concept*, in: Proceedings of the Annual Meeting of the Cognitive Science Society, 33, 2011.

[100] Ziauddin S., Dailey M.: *Iris recognition performance enhancement using weighted majority voting*, in: Proceedings / ICIP International Conference on Image Processing, 277 - 280, 2008.

[101] Żurawski M.: *Algorithms of induction of decision rules for multicriteria and multiattribute decision problems*, MSc thesis, 2001.