ПОНЯТИЕ В КОГНИТИВНО-ЛОГИЧЕСКОМ КОНТЕКСТЕ

Concept in a Cognitive-Logical Context

Д. Зайцев

30 апреля 2018 г.

МГУ имени М.В.Ломоносова, Москва, Россия

At present, the nature of conceptual structure remains unsettled.

Research into the nature of concepts is ongoing, in both philosophy and psychology, and there is no general consensus in either field as to the preferred theory of concepts.

At present, the nature of conceptual structure remains unsettled.

Research into the nature of concepts is ongoing, in both philosophy and psychology, and there is no general consensus in either field as to the preferred theory of concepts.

«Остается невыясненным основное: что представляет собой понятие как форма мысли и тем более как форма мышления вообще.»

At present, the nature of conceptual structure remains unsettled.

Research into the nature of concepts is ongoing, in both philosophy and psychology, and there is no general consensus in either field as to the preferred theory of concepts.

«Остается невыясненным основное: что представляет собой понятие как форма мысли и тем более как форма мышления вообще.»

Войшвилло 1989b, с. 87

Functions of Concepts:

- categorization (typification), recognition
- communication (understanding and comprehension)

ontology

A concept is

- abstract entity
- mental representation
- causal explanatory schema

stucture

A concept is

- definitional construction (genus-species)
- similarity construction
- theory
- unstructured primitive

The most influential theories of concepts

- **The Classical Theory**
  - abstract entity
  - definitional construction

The most influential theories of concepts

- **The Classical Theory**
  - abstract entity
  - definitional construction
- **The Prototype Theory (Exemplar Theory)**
  - mental representation
  - similarity construction

The most influential theories of concepts

- **The Classical Theory**
  - abstract entity
  - definitional construction
- **The Prototype Theory (Exemplar Theory)**
  - mental representation
  - similarity construction
- **The Theory-Theory**
  - causal explanatory schema
  - theory
- **Conceptual Atomism**
  - abstract entity
  - unstructured primitive

The Classical Theory in focus

- thoughts as abstract objects (Fregean senses)

The Classical Theory in focus

- thoughts as abstract objects (Fregean senses)
- thoughts as mental constructions (Voishvillo, Bocharov, Markin)

The Classical Theory in focus

- thoughts as abstract objects
  (Fregean senses)
- thoughts as mental constructions
  (Voishvillo, Bocharov, Markin)
- functions
  (Frege)

Concept as Function

"A concept is [a one-place] function whose value is always a truth value"

Concept as Function

"A concept is [a one-place] function whose value is always a truth value"

"Hence we can designate as an extension the value-range of a function whose value for every argument is a truth-value."

G. Frege "On Function and Concept".

Concept as Function

"A concept is [a one-place] function whose value is always a truth value"

"Hence we can designate as an extension the value-range of a function whose value for every argument is a truth-value."

G. Frege "On Function and Concept".

Frege would say that any object that a concept maps to The True falls under the concept. Thus, the number 2 falls under the concept 'prime number'.
Hence, a simple predication like '2 is prime' becomes analyzed in Frege's system as a special case of functional application.

Categorization

"Categorization is not a matter to be taken lightly. There is nothing more basic than categorization to our thought, perception, action, and speech."

(Lakoff, 1987, p. 5)

Categorization

”Categorization is not a matter to be taken lightly. There is nothing more basic than categorization to our thought, perception, action, and speech.”

(Lakoff, 1987, p. 5)

Typically in cognitive psychology, categorization is regarded as a process of determining what things ”belong together,” and a category is a group or class of stimuli or events that so cohere.

Thus, categorization is the mental operation by which the brain classifies objects and events.

Perceptual Categorization

Perceptual categorization is fundamental automatic part of the brain's remarkable ability to process large amounts of sensory information and efficiently recognize objects including speech.

It is the neural bridge between lower-level sensory and higher-level language processing.

Two Stages of Perceptual Categorization

The process of perceptual categorization is not homogeneous and can be very roughly divided into two phases, in accordance with Edelman's 'reentry' and Ivanitsky's 'informational synthesis':

Two Stages of Perceptual Categorization

The process of perceptual categorization is not homogeneous and can be very roughly divided into two phases, in accordance with Edelman's 'reentry' and Ivanitsky's 'informational synthesis':

1 – primary sensations recognition;

2 – subsequent synthesis of a perceived object as a whole.

> *Objects and their properties are perceived to be unitary, despite the fact that a given perception results from parallel activity in the brain of many different maps, each with different degrees of functional specialization. A striking case is the extrastriate visual cortex, in which different areas specialized for color, motion, and form act together to yield a coherent response to an object.*
>
> *(Edelman, 1993)*

Categorization as Falling under Concept

Both stages (low-level and high-level) of categorization can be considered as procedures of falling under (functional) concepts:

1 – each side, moment or part of perceived stimulus is falling under a corresponding concept of side, moment or part (shape, color, taste, etc.);
2 – the aggregation of typified components as a complex whole is falling under a concept corresponding to an intended object.

Intentional Theory of Concepts

The functional interpretation of concepts suggested below is a consequence of our to cognitive activity presented in:

2016 D. Zaitsev, N. Zaitseva 'Categorization in Intentional Theory of Concepts', Advances in Neural Networks – ISNN 2016, Lecture Notes in Computer Science, Vol. 9719, 465–473

2017 N. Zaitseva, D. Zaitsev 'Phenomenological Perspective for Moder Neiroscience' (in Russian), Russian Journal of Philosophical Sciences, 1, 71–84

Intentional Theory of Concepts – Motivation

Our approach to cognitive activity rests upon the following guiding principles:

1. We consider Intentionality to be a universal fundamental characteristics of cognition shared by animate bodies of various kinds.
2. Intentionality may be presented as functional relations from stimuli (taken as intended objects) to recognized individuals, relativized to a particular subject.
3. Understood that way, intentionality may be considered as a concept function from stimuli into intentional objects.

   In particular, as a consequence of this approach a categorization process was modeled via analogizing apperception-like function.

Apprensentation

The idea of apperceptive transfer, or analogizing appresentation, appears, when Husserl runs into a problem of Alter Ego in the Fifth Cartesian Meditation while trying to avoid charges of being prone in solipsism.

Literally, "appresentation" means making something "co-present". Due to analogizing apperception, Husserl shows that the Other is always a projection of my very self.

And not only the other self but any object of the world is typified "by analogy" with model object, earlier experienced by a cognitive agent.

Appresentation

Analogizing appresentation in turn is based on a more fundamental bottom procedure of pairing.

Pairing appears to be a form of passive synthesis. The idea behind it is that two objects are given in a phenomenological unity of similarity, which constitutes a pair. In this pair, the objective senses overlay, which results in a "mutual transfer of sense".

Husserl's famous example with scissors: A child, who has finally grasped the idea of scissors (understood "the final sense of scissors"), "sees scissors at the first glance as scissors".

(Husserl CM, p. 111).

Appresentation

Apperceptive transfer is neither an inference from analogy (and not an inference at all), nor a thinking act.

It is “a universal phenomenon of the transcendental sphere”, embedded and embodied fundamental cognitive mechanism, forming the basis of cognitive faculty as a directed interaction between a subject and an object.

In Husserl’s words: "Even the physical things of this world that are unknown to us are, to speak generally, known in respect of their type".

(Husserl CM, p. 111).

Typed Lambda Calculus

I propose a formal presentation of thus interpreted categorization by means of modified typed lambda calculus. More precisely categorization corresponds to beta-conversion

Typed lambda calculus is formal system (calculus) designed to implement a process of computation via application of functional abstractions. Its typed version is sensitive to a type of computed data.

Formally, it is based on two operations: lambda-abstraction λ and application $\bullet$.

Typed Lambda Calculus

If $M(x)$ means that Mx depends on x, then $\lambda x.M(x)$ is a functional abstraction expression denoting a map $x \longmapsto M(x)$.

Together application (of a function to an argument) and abstraction are enough to present a simple calculation, as illustrated by the following example: $\lambda x.(x+1) \bullet 2 = (2+1) = 3$.

In general the form of such calculation can be expressed as $\lambda x.M(x)) \bullet N = M(N)$, which is known as β-conversion.

Concept as (Relevant) Function

The straightforward idea is to consider *relevant propositional functions*, where
'*relevant*' applied to function means 'really depend on its argument',
and *propositional functions* correspond to predicates taking individuals as their arguments and returning truth-values.

Thereafter, a concept as a function when applied to individuals from its extension takes the True as its value.

Formal Presentation

Formal language:
- variables $\{V\}$,
- constants $\{K\}$,
- type-variables $\{\tau\}$
and three special symbols of
application $(\bullet)$,
lambda-abstraction λ and
functional mapping $(\longrightarrow)$.

Type and term (t) are defined as follows:
$\tau ::= p_i \mid A \longrightarrow A, \quad$ provided $p_i \in \{\tau\}$
$t ::= v \mid k \mid \lambda v.t \mid t \bullet t$

Formal Presentation

If t is a term and A – type, $t : A$ is a formula.

Formal Presentation

If t is a term and A – type, $t : A$ is a formula.

To increase visibility, for a moment I will distinguish between functional (predicate) constants and variables ($\mathfrak{F}$ and F correspondingly) and individual constants and variables ($\mathfrak{t}$ and v correspondingly).

Thus, in particular,
$\lambda v.\mathfrak{F} \bullet v : A \to B$, and
$\lambda F.F \bullet \mathfrak{t} : A \to B$
are concept-formulae.

Natural Deduction

Basic 'typed' rule of application:

$$AR \qquad \frac{t_1{:}A{\rightarrow}B\ \ [\Gamma] \quad t_2{:}A\ \ [\Delta]}{t_1{\bullet}t_2{:}B\ \ [\Gamma,\Delta]}$$

Abstraction:

$$AB \qquad \frac{t\ :\ B \quad [\Gamma,v{:}A]}{\lambda v.t{:}A{\rightarrow}B\ \ [\Gamma]}$$

Natural Deduction

Two rules of permutation:

$$PerB \qquad \frac{(\lambda v.k \bullet v) \bullet t_2 : B \quad [\Gamma]}{k \bullet (v \bullet t_2) : B \quad [\Gamma]}$$

$$PerC \qquad \frac{(\lambda v.v \bullet k) \bullet t : B \quad [\Gamma]}{(v \bullet t) \bullet k) : B \quad [\Gamma]}$$

Natural Deduction

Identity, or apperception rule:

$$IR \qquad \frac{t_1 \bullet (v \bullet k) \bullet t_2 : B [\Gamma]}{t_1 \bullet k \bullet t_2 : B \qquad [\Gamma]}$$

Examples

№1 'Property' typing schema

1. $\lambda V.V \bullet a : A \rightarrow B \; [\Gamma]$ *abstract concept*

Examples

№1 'Property' typing schema

1. $\lambda V.V \bullet a : A \to B \; [\Gamma]$ *abstract concept*
2. $F : A \; [\Delta]$ *side of an object (stimulus)*

Examples

№1 'Property' typing schema

1. $\lambda V.V \bullet a : A \to B \; [\Gamma]$ *abstract concept*
2. $F : A \; [\Delta]$ *side of an object (stimulus)*
3. $\lambda V.V \bullet a \bullet F : B \; [\Gamma, \Delta]$ AR, 1,2

Examples

№1 'Property' typing schema

1.	$\lambda V.V \bullet a : A \to B\ [\Gamma]$	*abstract concept*
2.	$F : A\ [\Delta]$	*side of an object (stimulus)*
3.	$\lambda V.V \bullet a \bullet F : B\ [\Gamma, \Delta]$	AR, 1,2
4.	$(V \bullet F) \bullet a : B\ [\Gamma, \Delta]$	$PerC$, 3

Examples

№1 'Property' typing schema

1. $\lambda V.V \bullet a : A \to B\ [\Gamma]$ *abstract concept*
2. $F : A\ [\Delta]$ *side of an object (stimulus)*
3. $\lambda V.V \bullet a \bullet F : B\ [\Gamma, \Delta]$ AR, 1,2
4. $(V \bullet F) \bullet a : B\ [\Gamma, \Delta]$ $PerC$, 3
5. $F \bullet a : B\ [\Gamma, \Delta]$ IR, 4

Examples

№2 Object typing schema

1. $\lambda x.F \bullet x : A \to B\ [\Gamma]$ *concrete concept*

Examples

№2 Object typing schema

1.	$\lambda x.F \bullet x : A \to B\ [\Gamma]$	*concrete concept*
2.	$a : A\ [\Delta]$	*object (stimulus)*

Examples

№2 Object typing schema

1.	$\lambda x.F \bullet x : A \to B\ [\Gamma]$	*concrete concept*
2.	$a : A\ [\Delta]$	*object (stimulus)*
3.	$\lambda x.F \bullet x \bullet a : B\ [\Gamma, \Delta]$	AR, 1,2

Examples

№2 Object typing schema

1. $\lambda x.F \bullet x : A \rightarrow B\ [\Gamma]$ *concrete concept*
2. $a : A\ [\Delta]$ *object (stimulus)*
3. $\lambda x.F \bullet x \bullet a : B\ [\Gamma, \Delta]$ AR, 1,2
4. $F \bullet (x \bullet a) : B\ [\Gamma, \Delta]$ $Per B$, 3

Examples

№2 Object typing schema

1.	$\lambda x.F \bullet x : A \to B \ [\Gamma]$	*concrete concept*
2.	$a : A \ [\Delta]$	*object (stimulus)*
3.	$\lambda x.F \bullet x \bullet a : B \ [\Gamma, \Delta]$	AR, 1,2
4.	$F \bullet (x \bullet a) : B \ [\Gamma, \Delta]$	$PerB$, 3
5.	$F \bullet a : B \ [\Gamma, \Delta]$	IR, 4

Quantifiers

The first option – λ-terms as denoting relations

On the relational reading, we can understand an application term $M \bullet N$ as a form of predication.

Then β-convertion $\lambda x.M \bullet a = M[x := a]$

says that the abstraction relation $\lambda x.M$, predicated of a, is the relation obtained by plugging in a for all free occurrences of x inside M.

Quantifiers

The first option – λ-terms as denoting relations

- John loves Mary: $loves(JOHN, MARY)$
- The property that John loves Mary: $\lambda.loves(JOHN, MARY)$
- The property of an object x that John loves it: $\lambda x.loves(JOHN, x)$.
- The property that Mary is loved by someone: $\lambda.\exists x(loves(x, MARY))$.

Quantifiers

The second option – Generalized Quantifiers

Expressions of type e denote elements of the universe of discourse.
Expressions of type t denote a truth value $\{t, f\}$.
Expressions of type $\langle e, t\rangle$ denote functions from the set of entities to the set of truth values.

We can now assign types to the words in our sentence 'Every boy sleeps' as follows.
Type(boy)= $\langle e, t\rangle$
Type(sleeps)= $\langle e, t\rangle$
Type(every)= $\langle\langle e, t\rangle, \langle\langle e, t\rangle, t\rangle\rangle$

Quantifiers

The second option – Generalized Quantifiers

Thus, 'every' denotes a function from a set to a function from a set to a truth value. It is that function which for any two sets A, B, every(A)(B)= 1 ⇔ A⊆B.

We can now write the meaning of 'every' with the following lambda term, where X, Y are variables of type $\langle e, t \rangle$:
$\lambda X, \lambda Y.X \subseteq Y$

Quantifiers

The second option – Generalized Quantifiers

Let 'boy' and 'sleeps' be 'B' and 'S', then for 'every boy sleeps' it holds that

$\lambda X.\lambda Y.\ X \subseteq Y \bullet B \bullet C$

$\lambda Y.\ B \subseteq Y \bullet C$

$B \subseteq S$

Quantifiers

The third option – polymorphic typed λ_2-calculus (Girard–Reynolds)

Quick intuition on system λ_2 – the introduction of a mechanism of universal quantification over types.

$$\forall\text{-elim} \qquad \frac{t \;:\; \forall v.A}{t\bullet k \;:\; A\ [v{:=}k]}$$

$$\forall\text{-intro} \qquad \frac{t \;:\; A}{\Lambda v.t \;:\; \forall v.A}$$

СПАСИБО ЗА ВНИМАНИЕ