TECHNICAL CONTRIBUTION

Semantics of Analogies from a Logical Perspective

Ahmed M. H. Abdelfattah[1] · **Ulf Krumnack**[2]

Abstract

A number of different approaches to model analogies and analogical reasoning in AI have been proposed, applying different knowledge representation and mapping strategies. Nevertheless, analogies still seem to be hard to grasp from a formal perspective, with no known treatment in the literature of, in particular, their formal semantics, though the empirical treatments involving human subjects are abundant. In this paper we present a framework that allows to analyze the syntax and the semantics of analogies in a universal logic-based setting without committing ourselves to a specific type of logic. We show that the syntactic process of analogy-making by finding a generalization can be given a sensible interpretation on the semantic level based on the *theory of institutions*. We then apply these ideas by considering a framework of analogy-making that is based on classical first-order logic.

1 Introduction

1.1 Motivation: The Core Cognitive Ability

Reasoning based on analogies is one of the most fundamental cognitive abilities to human thought [1] and, arguably, to some non-human cognitive beings [2]. This cognitive ability allows a reasoner to selectively retrieve needed information from the reasoner's memory or knowledge base, based on a matched situation to the reasoner's current situation. Analogies play a significant key role in a wide range of problem-solving contexts in cognitive science, with analogical reasoning providing a heuristic way of reasoning that differs from ways of exact reasoning. This facilitates the drawing of inferences that the reasoner thinks are specifically related to the current circumstances, because the very idea of making analogies depends on previously encountered experiences (e.g. the knowledge available in, or retrieved from, the reasoner's memory) and not on deterministic rules. Furthermore, even from a more abstract perspective, the establishment of analogical relations seems to be one of the

rare possibilities to explain many cognitive phenomena in a uniform way: quite often we act (alternatively perceive, reason, learn, etc.) as if we were in another (well-known and analogous) situation. It rarely happens that humans can reason in a purely deductive (abductive, inductive, etc.) way to act in real life. A natural description of such cognitive phenomena can be provided by analogies, because vagueness, learning, and the transfer of knowledge about old situations to new ones are intrinsically embedded in the very idea of analogy making.

Analogies attracted the attention of several cognitive science groups over the last few decades [2], especially those working in the artificial intelligence (AI) field. Extensive cognitive scientific research on analogy has been conducted, focusing on the central importance of analogical reasoning in diverse areas like perception, learning, memory, language, and thinking. Acts of remarkable intelligence and other cognitive abilities dot not only comprise (abstract) types of reasoning, learning from sparse data, and planning, but also the ability to solve problems based on similar solutions for other problems and to creatively finding new conceptualizations of an unknown domain. Thus, analogies are of interest for AI and cognitive science because it is possible to explain and model the latter types of abilities, at least to a certain extent, with frameworks for analogical reasoning.

Frameworks for analogy-making cover a large part of cognitive abilities that are usually considered to be central for modern AI systems. A good source to support this claim is [3], where analogies are used to model aspects of reasoning,

✉ Ahmed M. H. Abdelfattah
  ahabdelfattah@sci.asu.edu.eg

  Ulf Krumnack
  krumnack@uos.de

1 Faculty of Science, Ain Shams University, Cairo, Egypt

2 Institute of Cognitive Science, University of Osnabrück, Osnabrück, Germany

creativity, problem solving, learning, perception, or motor control, just to mention a few of them. Computational modeling of analogy making provides valuable sources of insights that led to a deeper understanding of analogy and the roles it plays in human cognition [4]. Approaches for deductive and inductive reasoning are well-examined in AI for decades, whereas analogical reasoning seems to be a hard challenge for machine intelligence despite the fact that analogies are considered a central mechanism of human cognition and intelligence. A number of models have been proposed to explain different aspects of analogies, varying in complexity and degree of formalization. Examples of such frameworks are the structure mapping engine (SME, which is probably the prototypical symbol-based analogy-engine) [5], interactionism [6], LISA [7], Copycat [4], and AMBR / DUAL [8]. An overview of various theoretical and practical approaches for the modeling of analogy making can be furthermore found in [1].

1.2 Importance: The Core Challenge

On the one hand, forms of analogical reasoning have always had the potential to be formalized and modeled. There are at least two different modes[1] of "analogies" that can be interpreted and computed: proportional and predictive analogies. There are also variations of the exact roles played by analogical reasoning in cognitive scientific studies. Various approaches or systems for computing analogies already exist, particularly using standard, logic-based mechanisms of knowledge representation and reasoning (KRR). On the other hand, and in spite of the broad variety of both syntactic and algorithmic core mechanisms, as well as computational models that have been proposed for analogy making, no similar standard mechanisms have been developed to provide a model-theoretic semantics of analogical relations. In fact it is hard to find uniform frameworks for the variety of different reasoning types and problem solving approaches in unknown situations [9, 10], or a spelled out semantics of analogical relations computed by algorithmic models. There is no uncontroversial theory of the semantics of analogies (although several models for 'computing' analogies have been proposed). Even worse, for all established frameworks even an endeavor to find a semantics of the underlying computations cannot be found. Clearly it is possible to formulate a denotational semantics based on term algebras for algebra-inspired approaches of analogical reasoning [6, 9] or for models using pattern matching methods [5]. But even if such ideas had been explicitly developed, it would remain unclear

how the semantic meaning of dynamically established analogical relations can emerge from these approaches.

As a consequence of these deficiencies, the desired generalization capabilities of appropriate cognitive systems in AI are currently far from being reachable. Yes; existing models show that various forms of analogy-based thinking are amenable to specification and formalization in cognitive systems, and that many elements of analogy-making can be characterized, formalized, and modeled. Yet, from an AI perspective, it is still desirable to have at least a model that could be combined with those standard mechanisms of KRR used in the cognitive system.

1.3 Significance: A Solution Perspective

The substantial significance of the work presented in this paper is to address the wide gap in treating analogies between AI and cognitive science, and to provide an initial step to formally equip logic-based frameworks of analogy-making with formal semantics. To contribute to bridging the gap between algorithmic approaches for analogical reasoning and the denotational semantics underlying these algorithms, the usage of analogy mechanisms for AI systems is proposed in this paper in an abstract way. Semantic issues of analogical relations are investigated and a model theory of analogical transfers is specified.

The presented approach is based on the *theory of institutions* [11]. We show that the particular syntactic process of analogy-making, in which a 'generalization' is found, can be given a sensible interpretation on the semantic level. Given models of source and target domains in an analogy-making process, the construction of models for the generalized theory of source and target will be examined. Furthermore, we specify some properties of both, the corresponding models and the underlying analogical relation.

The rest of the paper has the following structure. In the next section, Sect. 2, the modeling of abstraction-based analogies is explained in detail (Sect. 2.1), and then a formalization of our general modeling framework is presented (Sect. 2.2) based on this particular kind of modeling. A specific example of an analogy engine that can be slightly adapted to instantiate the described general modeling framework is outlined in Sect. 3 before we conclude with important remarks in Sect. 4.

2 A Perspective for the Semantics of Analogies

This section constitutes the major part of this paper. In Sect. 2.1, we first present and discuss a formal framework for modeling analogies based on a particular, rather standard, view of analogies adopted in the literature. This presentation

---

[1] In [9], an uncommon, third mode is presented, for which a computational model is difficult to implement.

is done on an abstract level and may be realized using different logical systems as we will see in Sect. 3. In the subsequent Sect. 2.2, we will spell out the formal framework of Sect. 2.1 for the case of classical predicate logic.

2.1 Modeling Analogies: A Standard, Informal Setting

The literature in cognitive science provides several theories and experiments on analogical reasoning [12]. Whilst earlier models tended to understand the basic constraints that govern human analogical thinking, the predominant objectives of recent theories have become to uncover psychological mechanisms of the sub-processes involved in analogy making, and to model the functioning of these sub-processes [2]. The experiments aim to support the theories by reporting on either (i) completing 'presented analogies', in which parts of analogies are given to human participants and are asked to complete these parts, or (ii) finding 'spontaneous analogies', where the participants initiate and form the entire analogy [13].

When the modeling of analogy-making as a cognitive process is considered, it is traditional to decompose it into multiple 'abstract processes' that reflect the stages of computing analogical reasoning, inspired by characterizing cognitive phases of analogical thinking. To model analogies by means of AI in cognitive systems, important processes need to be considered and described, such as analogy-making, transfer of knowledge, analogical reasoning and the like.

Our approach aims at achieving an abstract syntactic and semantic representation of modeling analogies in an arbitrary logical framework. By an 'arbitrary logical framework' we mean a general KRR framework that represents knowledge using a logic-based language (e.g. predicate first-order logic). We introduce our ideas on a intuitive level in this subsection and then formalize it in the following one.

Analogy-making can be broadly characterized by the result of finding an analogical relation linking salient 'conceptual entities' of two (structured) domains to each other. The first of these domains is designated as the "source" domain (the one which is standardly considered to be the richer domain including more available and accessible knowledge) and the other as the "target". Analogical reasoning is then seen as the ability to treat the target as being similar, in one aspect or another, to the source, depending on their shared commonalities in relational structure or appearance.

There seems to be a generally non-controversial core interpretation of analogies in the literature: Analogical relations can be established between a well-known domain, the source, and a formerly unknown domain, the target, without taking much input data (examples) into account. It is rather the case that a conceptualization of the source domain is
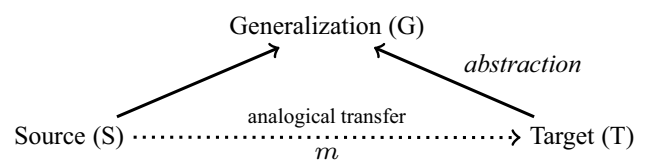

Fig. 1 An overall view of creating analogies

sufficient to generate knowledge about the target domain, which can be achieved by associating attributes and relations of the source and target domains. New "conceptual entities" can be productively introduced in the target domain by the projection (or the transfer) of attributes and relations from the source to the target, which allow the performance of reasoning processes to take place in the target domain.

The basic idea in our approach is to view analogy-making as an "abstraction process" of the input domains, along with relational mappings between corresponding 'conceptual entities' in the domains. That is, analogical relations between the entities in the input domains result from analogical transfer, which identifies their common (structural) aspects, and exports some of these aspects from the source to the target, in order for the treatment of the target (as being similar to the source) to consistently come about. The common aspects of the input domains are identified, and made explicit, in a generalized domain that captures the common parts of the input domains. This generalized domain can be thought of as being the mutual "generalization" sub-domain of both input domains. Figure 1 depicts this overall idea using S and T as source and target domains, respectively, where m represents an analogical relation between them, and the generalization, G, representing the common parts of S and T.

A prototypical example of analogy-making is the well-known one between the solar system and the Rutherford atom model, typically stated as: "the atom is like our solar system" [5]. In this example, the source domain simplifies a description of the solar system, where a planet revolves around the sun because the differences in mass result in different gravitational forces. The target domain describes the Rutherford atom model, in which the Coulomb force results in lightweight electrons being attracted by the nucleus that they also revolve around. A generalized domain of the two input domains can be obtained by matching knowledge entities and generalizing. For example, representations in the two domains may contain predicates in first-order logic that look like: $\forall(t : time) : \texttt{gravity}(\texttt{SUN}, \texttt{PLANET}, t) > 0$ and $\forall(t : time) : \texttt{dist}(\texttt{SUN}, \texttt{PLANET}, t) > 0$ (in the source); and $\forall(t : time) : \texttt{coulomb}(\texttt{NUCLEUS}, \texttt{ELECTRON}, t) > 0$ and $\forall(t : time) : \texttt{dist}(\texttt{NUCLEUS}, \texttt{ELECTRON}, t) > 0$ (in the target). These can be matched and generalized. Also, in each of the domains, there is an object that is revolving

around another, because of the difference in a specific property that both objects have, possibly captured by:

$\forall (t : time)(o_1, o_2 : object) : 0 < \mathtt{mass}(o_1) < \mathtt{mass}(o_2)$
$\wedge \mathtt{dist}(o_1, o_2, t) > 0 \wedge \mathtt{centrifugal}(o_1, o_2, t) < 0$
$\rightarrow \mathtt{revolves\_around}(o_1, o_2).$

After generalization is made, the reasoning concludes that the electrons and the nucleus keep a distance from each other (i.e., new knowledge in the target domain), just like the planets are distant from the sun (i.e., original knowledge in the source domain). This analogy situation illustrates several aspects of analogy-based reasoning, by comparing knowledge representations about the solar system, on the one hand, with representations (from a different field of knowledge) about the Rutherford atom model, on the other hand.

It is vital in this aspect to note that the abstraction process results in a generalization of the source and target, whilst the inverse operation; namely, to construct the input domains from the generalization, can still be modeled by 'substitutions', which result in specializations of the generalization. Obviously, making a 'substitution' is not a one-to-one inverse process of making a generalization, as there are many possibilities to associate entities of two domains resulting in different candidates of analogies. Furthermore, analogies are not considered correct or incorrect, but more or less psychologically plausible. In more formal terms (cf. Sect. 2.2), this can be assessed by a heuristic governing the trade-off between the degree to which the generalization 'covers' both input domains and the complexity of the generalization. It is worth mentioning that the amount of 'coverage' in analogy-making plays a role, since it basically reflects the limit to which parts of the domains may or may not be included in the generalization.[2] Moreover, not only some parts of the domains may be irrelevant to each other, of course, but also the domains can be pragmatically irrelevant for a human reasoner. The earlier discussions informally explain in plain language what is formally captured in abstract symbols below.

2.2 Modeling Analogies: A Formalization

As mentioned in Sect. 2.1, the generalization of the source and target results from an abstraction process, for which the inverse operation is to construct the input domains by 'substitutions'. Therefore, we will use the language of "institution theory" [15] to capture this aspect and formally present our ideas. Institutions formalize the intuitive notion of logical systems, including both syntax and semantics [11], by using methods from category theory. We start by introducing the basic concepts before turning towards analogies. The notion of institutions was originally introduced in the 1970s and has been continuously elaborated since then. For a gentle introduction to the theory of institutions see [15]. For a more recent and comprehensive writing the interested reader is referred to [16].

Definition 1 *An institution* $\mathbb{I} = \langle \mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \vDash_\Sigma \rangle$ consists of:

1. a category **Sign** of signatures,
2. a functor $\mathbf{Sen} : \mathbf{Sign} \rightarrow \mathbf{Set}$ mapping each signature Σ to a set of sentences $\mathbf{Sen}(\Sigma)$ and each signature morphism $\sigma : \Sigma \rightarrow \Sigma'$ to the sentence translations map $\mathbf{Sen}(\sigma) : \mathbf{Sen}(\Sigma) \rightarrow \mathbf{Sen}(\Sigma')$,
3. a functor $\mathbf{Mod} : \mathbf{Sign^{op}} \rightarrow \mathbf{CAT}$ mapping each signature Σ to the category of models $\mathbf{Mod}(\Sigma)$ and each signature morphism $\sigma : \Sigma \rightarrow \Sigma'$ to the reduct functor $\mathbf{Mod}(\sigma) : \mathbf{Mod}(\Sigma') \rightarrow \mathbf{Mod}(\Sigma)$, and
4. a relation $\vDash_\Sigma \subset |\mathbf{Mod}(\Sigma)| \times \mathbf{Sen}(\Sigma)$ such that for each $\sigma : \Sigma \rightarrow \Sigma'$ the following satisfaction relation holds: $\mathfrak{m}' \vDash_{\Sigma'} \mathbf{Sen}(\sigma)(\rho)$ if and only if $\mathbf{Mod}(\sigma)(\mathfrak{m}') \vDash_\Sigma \rho$ for each $\mathfrak{m}' \in |\mathbf{Mod}(\Sigma')|$ and $\rho \in \mathbf{Sen}(\Sigma)$.

A simple example of an institution can be given in well-known terms of first-order logic (FOL). In this case, the Σ-sentences $\mathbf{Sen}(\Sigma)$ corresponds to the set of all FOL formulas that can be built using symbols from a signature Σ. For each signature Σ the collection $\mathbf{Mod}(\Sigma)$ of all Σ-models corresponds in FOL to the collection of all possible interpretations of symbols from Σ. The Σ-models and Σ-sentences are related by the relation of Σ-satisfaction, which corresponds to the classical model theoretic satisfaction relation in FOL.

In institution theory, signature morphisms induce mappings on the sentence and the model class level. For our exposition, the notion of a signature morphism does not suffice, so we will make use of the wider concept[3] of general substitutions.
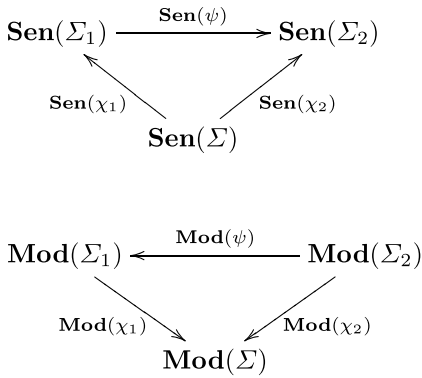
Definition 2 For a signature Σ of an institution, and a signature morphisms $\chi_1 : \Sigma \rightarrow \Sigma_1$ and $\chi_2 : \Sigma \rightarrow \Sigma_2$, a *general Σ-substitution* $\psi_{\chi_1 : \chi_2}$ consists of a pair $\langle \mathbf{Sen}(\psi), \mathbf{Mod}(\psi) \rangle$, where:

- $\mathbf{Sen}(\psi) : \mathbf{Sen}(\Sigma_1) \rightarrow \mathbf{Sen}(\Sigma_2)$ is a function,
- $\mathbf{Mod}(\psi) : \mathbf{Mod}(\Sigma_2) \rightarrow \mathbf{Mod}(\Sigma_1)$ is a functor,

such that both of them preserve Σ, i.e. the following diagrams commute:

---

[2] The reader can refer to [14] for a more exhaustive elaboration on this particular topic of coverage.
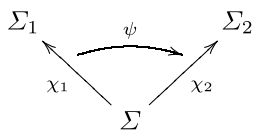
[3] This concept is based on [16, section 5.3].

$$\mathbf{Sen}(\Sigma_1) \xrightarrow{\mathbf{Sen}(\psi)} \mathbf{Sen}(\Sigma_2)$$

with $\mathbf{Sen}(\chi_1)$ and $\mathbf{Sen}(\chi_2)$ arrows from $\mathbf{Sen}(\Sigma)$.

$$\mathbf{Mod}(\Sigma_1) \xleftarrow{\mathbf{Mod}(\psi)} \mathbf{Mod}(\Sigma_2)$$

with $\mathbf{Mod}(\chi_1)$ and $\mathbf{Mod}(\chi_2)$ arrows to $\mathbf{Mod}(\Sigma)$.

and such that the following satisfaction condition holds:

$$\mathbf{Mod}(\psi)(\mathfrak{m}_2) \vDash \rho_1 \quad \text{if and only if} \quad \mathfrak{m}_2 \vDash \mathbf{Sen}(\psi)(\rho_1)$$

for each Σ_2-model $\mathfrak{m}_2$ and each Σ_1-sentence ρ_1.

A general Σ-substitution ψ from χ_1 to χ_2 can be indicated by a diagram of the following type:

$$\Sigma_1 \xleftarrow{\chi_1} \Sigma \xrightarrow{\chi_2} \Sigma_2 \quad \text{with } \psi$$
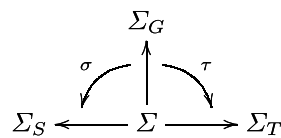
As can be seen, general Σ-substitutions extend the idea of a signature morphism. Although in general there needs to be no mapping on the level of signatures, most general Σ-substitution considered in practice are induced by some form of signature mapping. Every signature morphism can be seen as general Σ-substitution, and many other mappings, like classical first-order substitutions, second-order substitutions (for FOL), and derived signature morphisms give rise to a general Σ-substitution.

By developing our framework based on the abstract notions of institutions and general Σ-substitutions, we can express the essential ideas on a universal level and make clear the core notions without having to care for technical details. The framework captures the essential constraints that hold in a variety of logical systems, particularly those mentioned in Sect. 2.1. This will also provide useful properties for free, which may be difficult to formally prove for a specific formalism. It also allows for different mappings in 'implantation' (that can be chosen for a specific situation) based on heuristics or complexity constraints.[4]
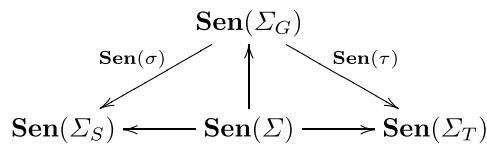
We now turn to modeling analogies in this framework. We assume that a source domain *S* and a target domain *T* are given using a logical formalism, i.e. within a common

institution 𝕀. The signatures for these domains are denoted by Σ_S and Σ_T respectively. We will further allow the use of common symbols from a signature Σ. We also assume the central idea that an analogy is mediated by a generalization that captures the common structures of source and target domain. Hence, an analogical relation can be described by a generalization that reflects the common structures in both domains (cf. Sect. 2.1). We introduce a further signature Σ_G that provides the generalized symbols. Now an analogy can formally be defined as follows:
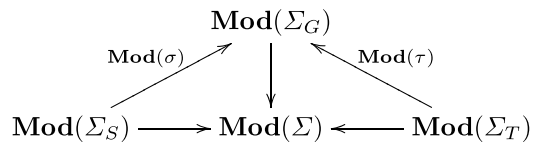
Definition 3 Given two signatures Σ_S and Σ_T over a common signature Σ, an *analogy* ℵ is defined to be a triple $\langle \Sigma_G, \sigma, \tau \rangle$, consisting of a signature Σ_G, and general Σ substitutions σ and τ as indicated in the following diagram:

$$\Sigma_S \xleftarrow{} \Sigma \xrightarrow{} \Sigma_T \quad \text{with } \sigma, \tau \text{ to } \Sigma_G$$

As a Σ-substitution is defined as a pair of mappings on sentence and model level, every analogy gives rise to the following diagrams:

$$\mathbf{Sen}(\Sigma_S) \xleftarrow{} \mathbf{Sen}(\Sigma) \xrightarrow{} \mathbf{Sen}(\Sigma_T)$$

with $\mathbf{Sen}(\sigma)$ and $\mathbf{Sen}(\tau)$ to $\mathbf{Sen}(\Sigma_G)$.

and

$$\mathbf{Mod}(\Sigma_S) \xrightarrow{} \mathbf{Mod}(\Sigma) \xleftarrow{} \mathbf{Mod}(\Sigma_T)$$

with $\mathbf{Mod}(\sigma)$ and $\mathbf{Mod}(\tau)$ from $\mathbf{Mod}(\Sigma_G)$.

Furthermore, for every Σ_G-sentence ρ, every Σ_S-model $\mathfrak{m}_S$ and every Σ_T-model $\mathfrak{m}_T$, the following satisfiability conditions hold:

$$\mathfrak{m}_S \vDash_{\Sigma_S} \mathbf{Sen}(\sigma)(\rho) \quad \text{iff} \quad \mathbf{Mod}(\sigma)(\mathfrak{m}_S) \vDash_{\Sigma_G} \rho$$

and

$$\mathbf{Mod}(\tau)(\mathfrak{m}_T) \vDash_{\Sigma_G} \rho \quad \text{iff} \quad \mathfrak{m}_T \vDash_{\Sigma_T} \mathbf{Sen}(\tau)(\rho)$$

In this setting, we can introduce an analogical relation on the level of sentences as well as on the level of models.

Definition 4 Given an analogy ℵ the associated *analogical relation* $\overset{\aleph}{\sim}$ is defined as a pair $\langle \overset{\aleph}{\sim}_{\mathbf{Sen}}, \overset{\aleph}{\sim}_{\mathbf{Mod}} \rangle$ of relations:

- On the level of sentences: for every $s \in \mathbf{Sen}(\Sigma_S)$ and $t \in \mathbf{Sen}(\Sigma_T)$, the relation $s \overset{\aleph}{\sim}_{\mathbf{Sen}} t$ holds iff there exists a $g \in \mathbf{Sen}(\Sigma_G)$ such that $\mathbf{Sen}(\sigma)(g) = s$ and $\mathbf{Sen}(\tau)(g) = t$.
- On the level of semantics: for every $\mathfrak{m}_S \in \mathbf{Mod}(\Sigma_S)$ and $\mathfrak{m}_T \in \mathbf{Mod}(\Sigma_T)$, the relation $\mathfrak{m}_S \sim_{\mathbf{Mod}} \mathfrak{m}_T$ holds iff $\mathbf{Mod}(\sigma)(\mathfrak{m}_S) = \mathbf{Mod}(\tau)(\mathfrak{m}_T)$.

A direct consequence from this definition gives the following facts.

Fact 1 *Let $\aleph$ be an analogy. For every pair of sentences $s \in \mathbf{Sen}(\Sigma_S)$, $t \in \mathbf{Sen}(\Sigma_T)$, and every pair of models $\mathfrak{m}_S \in \mathbf{Mod}(\Sigma_S)$, $\mathfrak{m}_T \in \mathbf{Mod}(\Sigma_T)$ with $s \overset{\aleph}{\sim}_{\mathbf{Sen}} t$ and $\mathfrak{m}_S \overset{\aleph}{\sim}_{\mathbf{Mod}} \mathfrak{m}_T$ it holds that $\mathfrak{m}_S \vDash s$ iff $\mathfrak{m}_T \vDash t$.*

Based on these notions, we now consider the theories, i.e. sets of sentences $\mathrm{Th}_S \subseteq \mathbf{Sen}(\Sigma_S)$ and $\mathrm{Th}_T \subseteq \mathbf{Sen}(\Sigma_T)$, used to model the source and target domain respectively. We say that a set of Σ_G-sentences, Th, is a $\aleph$-*generalization* of Th_S and Th_T, if $\mathbf{Sen}(\sigma)(\mathrm{Th}) \subseteq \mathrm{Th}_S$ and $\mathbf{Sen}(\tau)(\mathrm{Th}) \subseteq \mathrm{Th}_T$.

Fact 2 *For every analogy $\aleph$ and every pair of theories, Th_S and Th_T, there exists a maximal (with respect to set inclusion) $\aleph$-generalization Th_G.*

We will call Th_G the $\aleph$-*generalization of* Th_S *and* Th_T. The sentences $\mathbf{Sen}(\sigma)(\mathrm{Th}_G)$ and $\mathbf{Sen}(\sigma)(\mathrm{Th}_G)$ are said to be the parts of the domain that are *covered* by the analogy.

Fact 3 *For every sentence $s \in \mathbf{Sen}(\Sigma_S)$ that is covered by an anlogy $\aleph$, there exists a sentence $t \in \mathbf{Sen}(\Sigma_T)$ such that $s \overset{\aleph}{\sim}_{\mathbf{Sen}} t$ and vice versa.*

This concludes our modeling proposal of a general framework applicable to arbitrary[5] analogy-making approaches which are based on a broad range of underlying logical systems. As long as (i) common expressions in the source and target domains are associated via an analogy $\aleph$, (ii) a generalization for both domains is computed, and (iii) source and target can be recovered from the generalization using substitutions; then the analogy $\aleph$ can be formally described on the syntactic and semantic level.

3 HDTP as a Sample Framework

One of our aims is to model semantics independently of the particular logic used. This is achieved in the previous section using the abstract language of institutions, which

makes no assumption on the type of logic in which the analogical reasoning formalism is realized. But one also needs to investigate some characteristics that must be met in the frameworks to which one could apply the abstract ideas. We need at least a proof of concept. For example, some logic-based frameworks may not fully comply with the FOL institution as explained in Sect. 2, so we need to study how it could be possible to apply our ideas to such frameworks. This section presents an instance of a logic-based analogy-making system that shows there is at least a framework to which we can apply the abstract ideas.

Heuristic-Driven Theory Projection (HDTP) [1] is an example of an analogy engine that we choose in this paper to instantiate our general formal framework described in Sect. 2 specifically for first-order logic (FOL). We explain in the following why HDTP is a good candidate and how a slightly modified version of HDTP is a more suitable candidate for such an instantiation. Note that our abstract ideas could not be applied to any analogy-making framework in the original form, because a logic-based framework may need to satisfy further constraints in order to play the role of an institution that complies with the presented definitions. In its original version, HDTP provides an explicit generalization of two given domains specified as theories in (many-sorted) FOL as a by-product of establishing an analogy.[6] HDTP proceeds in two phases: in *the mapping phase*, the source and target domains are compared to find structural commonalities, and a generalized description is created, which subsumes the matching parts of both domains. In HDTP's *transfer phase*, unmatched knowledge in the source domain can be mapped to the target domain to establish new hypotheses. The overall idea of establishing an analogy between two domains in the HDTP framework is depicted in Fig. 1, which makes it a feasible candidate to apply general Σ-substitutions in the kind of analogies defined in Definition 3 (cf. Sect. 2.2).

For our current purposes, we will consider the mapping mechanism in more detail. The mapping is achieved via a generalization process, in which pairs of formulas from the source and target domain are anti-unified resulting in a generalized theory that reflects common aspects of the two domains. Formulas that are generalized to the same element in the generalized theory are considered to be analogically related. The generalized theory can be projected into the original domains by substitutions which are computed during anti-unification. A domain formula is said to be *covered* by the analogy, if it is within the image of this projection, otherwise it is *uncovered*. In analogy making, the analogical relation is used in the transfer phase to translate additional uncovered knowledge from the source to the target domain.

Technically, the original version of HDTP is based on restricted higher-order anti-unification [1] which is defined on four basic types of substitutions.[7] Therefore, and in order to allow a mild form of higher-order anti-unification in HDTP, one can extend classical FOL terms by introducing variables that can take arguments: for every natural number n we assume an infinite set $\mathcal{V}_n$ of variables with arity n. Here we explicitly allow the case $n = 0$ with $\mathcal{V}_0$ being the set of FOL variables. In this setting, a *term* is either a first-order or a higher-order term, i.e. an expression of the form $F(t_1, \ldots, t_n)$ with $F \in \mathcal{V}_n$ and terms $t_1, \ldots, t_n$. In addition, we also introduce the following set of basic substitutions.

Definition 5

1. A *renaming* $\rho^{F,F'}$ replaces a variable $F \in \mathcal{V}_n$ by another variable $F' \in \mathcal{V}_n$ of the same argument structure:

$$F(t_1, \ldots, t_n) \xrightarrow{\rho^{F,F'}} F'(t_1, \ldots, t_n).$$

2. A *fixation* ϕ_c^F replaces a variable $F \in \mathcal{V}_n$ by a function symbol f of the same argument structure:

$$F(t_1, \ldots, t_n) \xrightarrow{\phi_f^F} f(t_1, \ldots, t_n).$$

3. An *argument insertion* $\iota_{G,i}^{F,F'}$ with $0 \leq i \leq n$, $F \in \mathcal{V}_n$, $G \in \mathcal{V}_k$ with $k \leq n - i$, and $F' \in \mathcal{V}_{n-k+1}$ is defined by

$$F(t_1, \ldots, t_n) \xrightarrow{\iota_{G,i}^{F,F'}}$$
$$F'(t_1, \ldots, t_i, G(t_{i+1}, \ldots, t_{i+k}), t_{i+k+1}, \ldots, t_n).$$

4. A *permutation* $\pi_\alpha^{F,F'}$ with $F, F' \in \mathcal{V}_n$ and bijective $\alpha : \{1, \ldots, n\} \to \{1, \ldots, n\}$ rearranges the arguments of a term:

$$F(t_1, \ldots, t_n) \xrightarrow{\pi_\alpha^{F,F'}} F'(t_{\alpha(1)}, \ldots, t_{\alpha(n)}).$$

For generalizing complex terms, we can successively apply several substitutions: To receive a non-ambiguous set of substitutions we apply the basic substitutions in the order renaming, argument insertion, permutation, and finally fixation. We will call any composition of basic substitutions a (higher-order) substitution and write $t \to t'$, if there exists a sequence of basic substitutions that transforms t into t'. We will call t' an (higher-order) instance of t, and t an (higher-order) anti-instance of t'.

Now, after introducing the slight modifications we presented earlier in this section, HDTP can finally be interpreted using the concepts described in Sect. 2:

1. The underlying institution $\mathbb{I}$ is simply the institution FOL, i.e. the category of signatures **Sign** is the category of logical first-order signatures,
2. the collection of Σ-sentences is the class of all first-order formulas, and
3. the collection **Mod**(Σ) of all Σ-models is the collection of all possible interpretations of symbols from Σ.
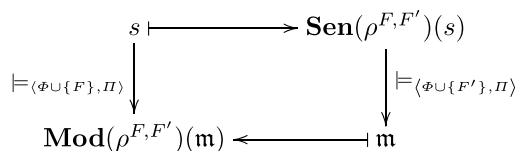
The basic substitutions from Definition 5 give rise to mappings on the syntactic and semantic level.[8] On the syntactic level, a basic substitution replaces function and predicate symbols in the way described above, inducing a function on the set of FOL formulas. For example, given a FOL signature $\Sigma = \langle \Phi, \Pi \rangle$ with function symbols Φ and predicate symbols Π, a renaming $\rho^{F,F'}$ induces a function

$$\mathbf{Sen}(\rho^{F,F'}) : \mathbf{Sen}(\Phi \cup \{F\}, \Pi) \to \mathbf{Sen}(\Phi \cup \{F'\}, \Pi)$$

On the semantic level, we get a mapping in the opposite direction: given a model for $\langle \Phi \cup \{F'\}, \Pi \rangle$, we can interpret $\langle \Phi \cup \{F\}, \Pi \rangle$-formulas, by first translating them via $\rho^{F,F'}$ and using the given model. Hence $\rho^{F,F'}$ induces a mapping of models

$$\mathbf{Mod}(\rho^{F,F'}) : \mathbf{Mod}(\Phi \cup \{F'\}, \Pi)$$
$$\to \mathbf{Mod}(\Phi \cup \{F\}, \Pi).$$

It is easy to see, that the satisfiability condition is fulfilled:

$$
\begin{array}{ccc}
s & \longmapsto & \mathbf{Sen}(\rho^{F,F'})(s) \\
\models_{\langle \Phi \cup \{F\}, \Pi \rangle} \Big\downarrow & & \Big\downarrow \models_{\langle \Phi \cup \{F'\}, \Pi \rangle} \\
\mathbf{Mod}(\rho^{F,F'})(\mathfrak{m}) & \longleftarrow\!\!\shortmid & \mathfrak{m}
\end{array}
$$

The same holds for the other basic substitutions as can be easily verified. Hence every HDTP substitution $t \to t'$, as a composition of basic substitutions resulting in an anti-instance t of t', gives rise to a general substitution in the sense of Definition 2. Therefore the operations HDTP performs in computing an analogical relation between a given source and target domain fit into the framework presented in Sect. 2. Furthermore, a syntactic and semantic interpretation of the HDTP computation is provided by the presented approach. This allows to adopt the notions of analogical relation and coverage introduced there (cf. the end of Sect. 2.1).

---

[7] Restricted higher-order anti-unification resembles to a certain extent strategies proposed in the context of ontology repair plans [17].

[8] In fact, the basic substitutions are special cases of second-order substitutions in first-order logic, which can also be described as derived signature morphisms, cf. [16, 101].

4 Conclusion

Analogies are considered a core mechanism for intelligence and cognition and have been extensively studied from a variety of perspectives. In cognitive science, analogies provide a model to describe a reasoning behavior, and can be used to explain cognitive abilities like learning from sparse data, creative problem solving, abstractions of concrete situations, and recognition of formerly unseen situations, just to mention a few examples. In artificial intelligence, different senses of computing analogies (e.g., proportional vs. predictive) as well as various analogy-making models have been given. There is an extremely huge gap, however, between the many, many accounts in cognitive science (of what analogies are and how they work, on the one side) and the many, many computational frameworks in AI (about how analogies can be made or found, on the other side).

We claim that this paper has the quality of being original in formally addressing this gap. This is a step in bridging the semantics with computations. There is no previous contribution in the literature of AI and cognitive science that addresses the semantics aspects of analogies, in particular for logic-based frameworks, though the challenge itself is as old as the modern, interdisciplinary studies in cognitive science. According to Fodor's modularity hypothesis, very global processes, like analogical reasoning, are not understood at all in cognitive science. Jerry Fodor explicitly states that "everybody thinks that analogical reasoning is an important ingredient in all sorts of cognitive achievements that we prize", yet he also states that "nobody knows anything about how it works" [18, pp. 107]. More broadly speaking, a central dogma in cognitive science, which dates back to its formative conferences in the mid-20th century, is that the mind is performing cognition and making decisions based on information processing, though there is always a tension between cognitive scientists on how exactly this processing is performed [19]. Hence, it is not surprising that providing at least one formal method for modeling the semantics of analogies must be of an extreme importance in bridging the gap between findings in cognitive science and AI approaches in modeling human reasoning tasks within automated reasoning systems, particularly in general logical frameworks.

The method presented here represents analogies both on the syntactic and the semantic level using the theory of institutions, which renders the method amenable to use in logic-based frameworks. As demonstrated, this seems to work in principle, and an application of this idea to one of the first-order logic frameworks for analogy-making, HDTP, is given as a proof of concept. Indeed, HDTP is perfectly suited for the adaptation (after slight modifications), but the application can in general be adapted in the future to work on other logic-based frameworks for analogy-making, perhaps also after introducing slight modifications to the original settings of that selected framework (in a way similar to what we followed with HDTP in this paper). Establishing an analogy between two input domains by an abstraction process does not only agree with the intuitions gained by observing cognitive processes, but can also be given a sensible interpretation on the semantic level, and implemented in logic-based AI systems. As the paper shows, the generalized theory and the established analogical relation (on a purely syntactic basis) can be connected to model theoretic relations on the semantic level in a coherent manner.

We believe that the formal analysis of the model theoretic semantics of analogies will be helpful for developing and improving computational models for (or based on) analogical reasoning. A better understanding of this type of representation can help in modeling certain types of creativity as well as understanding and explaining a broad range of cognitive capacities of humans, as discussed in e.g. [14]. Several interesting open questions remain. First, as we just mentioned, the framework should be applied to other symbolic analogy models, such as SME [5], for example. Second, several approaches in the field of theorem proving and ontology repair systems include substitutions resulting in a change of the underlying signature of the respective theory [17, 20]. Last but not least, the present paper sketches only the specification of the syntax and semantics of certain non-classical forms of reasoning in a first-order logic setting. A thorough examination of extensions of reasoning with respect to alternative logical systems (like higher-order logic, description logics, modal logic, equational logic) remains a topic for the future.

References

1. Schwering A, Kühnberger K-U, Kokinov B (2009) Analogies: integrating multiple cognitive abilities (guest editorial). Special Issue Anal Integr Cogni Abil J Cognit Syst Res 10(3):175–177
2. Bartha P (2013) Analogy and analogical reasoning. In: Zalta EN (ed) The Stanford encyclopedia of philosophy, fall 2013 edn. The Metaphysics Research Lab, Stanford University, Stanford
3. Kokinov B, Holyoak K, Gentner D (eds) (2009) New frontiers in analogy research. NBU Press, Sofia
4. Hofstadter D, the Fluid Analogies Research Group (1995) Fluid concepts and creative analogies. Computer models of the fundamental mechanisms of thought. Basic Books, New York
5. Falkenhainer B, Forbus K, Gentner D (1989) The structure-mapping engine: algorithm and example. Artif Intell 41:1–63
6. Dastani M, Indurkhya B, Scha R (2003) Analogical projection in pattern perception. J Exp Theor Artif Intell 15(4):489–511. https://doi.org/10.1080/09528130310001626283
7. Hummel JE, Holyoak KJ (1996) Lisa: a computational model of analogical inference and schema induction. In: Cottrell GW (ed) Proc. of the 18th annual conference of the cognitive science society, pp 52–357
8. Kokinov BN, Petrov AA (2001) Integration of memory and reasoning in analogy-making: the AMBR model. In: Gentner D et al

(eds) The analogical mind: perspectives from cognitive science. MIT Press, Cambridge, pp 59–124

9. Indurkhya B (1992) Metaphor and cognition: an interactionist approach. Studies in cognitive systems. Kluwer Academic, Dordrecht

10. Goertzel B, Pennachin C (2010) Artificial general intelligence. Cognitive technologies. Springer, Berlin

11. Goguen J, Roşu G (2002) Institution morphisms. Form Asp Comput 13(3–5):274–307

12. Gentner D, Holyoak K, Kokinov B (eds) (2001) The analogical mind: perspectives from cognitive science. MIT Press, Cambridge

13. Clement JJ (2008) Creative model construction in scientists and students: the role of imagery, analogy, and mental stimulation. Springer, London

14. Abdel-Fattah A, Besold T, Kühnberger K.-U (2012) Creativity, cognitive mechanisms, and logic. In: Bach J, Goertzel B, Iklé M (eds) Proc. of the 5th conference on AGI, Oxford, vol 7716 of lecture notes in computer science, pp 1–10

15. Goguen JA, Burstall RM (1992) Institutions: abstract model theory for specification and programming. J ACM 39:95–146

16. Diaconescu R (2008) Institution-independent model theory. Studies in universal logic. Birkhäuser, Basel

17. McNeill F, Bundy A (2007) Dynamic, automatic, first-order ontology repair by diagnosis of failed plan execution. Int J Semant Web Inf Syst 3:1–35

18. Fodor JA (1983) The modularity of mind. A Bradford book. MIT Press, Cambridge

19. Bermúdez J (2014) Cognitive science: an introduction to the science of the mind. Cambridge University Press, Cambridge

20. Chan M, Lehmann J, Bundy A (2010) Higher-order representation and reasoning for automated ontology evolution. In: Proceedings of the international conference on knowledge engineering and ontology development, pp 84–93