A Computational Method for Philosophical Interpretation

MSc Thesis *(Afstudeerscriptie)*

written by

Silvan Hungerbühler
(born 26.10.1992 in Lenzburg)

under the supervision of **Arianna Betti** and **Veruska Zamborlini**, and submitted to the Board of Examiners in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:
06.07.2018

Members of the Thesis Committee:
Katrin Schulz
Arianna Betti
Veruska Zamborlini
Hein van den Berg
Luca Incurvati
Erman Acar

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Abstract

This thesis seeks to advance the use of computational techniques for the task of philosophical interpretation. To this end I present a novel method which draws on techniques and formal tools from logic-based knowledge modeling, automated reasoning, and natural language processing to support researchers in philosophy. This highly-interdisciplinary and experimental effort forms part of the e-Ideas project which seeks to build a new, computationally-scalable methodology for History of Ideas. I test my method on concrete research questions under investigation by e-Idea's members and thus provide an illustration of my method's functioning in practice as well as evidence for its methodological adequacy.

Contents

1 Introduction

The goal of this thesis is to contribute to the effort of making use of formal and computational tools to support academic philosophical research. Specifically, I present a method to support researchers working on philosophical interpretation. Philosophical *interpretation* or *exegesis* denotes the effort of arriving at a correct interpretation of a philosophical text. This is a central activity in philosophical practice because there is often a great deal of disagreement among scholars as to what precisely it is that a piece of philosophical writing expresses. I will be concerned with philosophy in its written form. Nothing necessarily confines philosophy to this medium, of course, but textual presentations certainly have been and still are the central locus of Western academic philosophical discourse.

The concrete contribution I present within this thesis is to combine a number of formal and computational techniques to build a method which can support researchers working on philosophical interpretation. I present a schematic outline of how tools developed for formal knowledge representation, automated reasoning, and text mining can be used to gather evidence for or against an interpretation of a primary philosophical text. This schematic outline is not meant to be completely specified. The steps and processes comprising this outline could be concretely realized in many different ways. To give an indication of the method's concrete implementation potential, I present excerpts of my own attempts of using the method together with a team of researchers. These cases of application give some illustration of how the method can be fully fleshed out. Simultaneously, they are also used to scaffold my argument for the method's adequacy.

1.1 Embedding Within the e-Ideas Project

This thesis forms part of an on-going effort by the e-Ideas project to provide a methodological basis for incorporating into philosophical research specific techniques developed for natural language processing[1] (NLP, for short) and logic-based modeling of expert knowledge about philosophical domains.[10] The e-Ideas project is based in Amsterdam and brings together philosophers and computational experts to combine a novel theoretical foundation for the history of ideas with the computational power of digital tools.[10][2] *History of ideas* refers to efforts to trace concepts through time and across locations. For this purpose, researchers in the project are currently devising a sound methodology that, although retaining the fundamental nature of traditional methods in the history of ideas, can be scaled up computationally to be used on a much larger quantity of data.[10]

A central premise of the project is that merely applying shallow bottom-up methods[3] from NLP to digitalized texts is insufficient for thorough philosophical

[1]For a very brief description of what NLP encompasses, see the opening paragraphs of Section 3.

[2]See this link for the project's website: https://conceptsinmotion.org/e-ideas/.

[3]By "Bottom-up methods" I refer here to strategies of information processing in which the analysis of the simplest parts of the data takes primacy. From the analysis of these simple parts an analysis of the whole set of data emerges. Typically, this takes the form of statistical methods where simple parts, such as words, form part of distributions. I call "Top-down approaches", on the other hand, strategies which first demand a formulation of an overview of the data which then guides the analysis of its sub-parts.

research. Used in isolation, such techniques fall woefully short of the requirements of extremely fine-grained and contextualized analyses of textual material as conceptually dense as philosophy tends to be. To circumvent this problem, e-Ideas rests on Betti's and Van den Berg's notion of *conceptual models*.[12] A conceptual model represents ideas as complex relational structures with both stable and variable elements.[12] Due to their high level of structure and the fact that they systematically make explicit the knowledge presupposed by humanities experts, these models are well-suited for implementation in a computational system. e-Ideas combines these formal representations of expert knowledge with automatic ontology extraction from the textual data.[10] The project attempts to strike the right balance between top-down framing of what is already known and bottom-up learning from the yet uninterpreted data. The hope driving this approach is that the computer will allow researchers access to evidence on an unprecedented scale which is based, nonetheless, upon sound methodology. In sum, e-Ideas seeks to combine two different computational approaches to study large corpora of text.[10]

The computational approaches are designed to supplement traditional philosophical methods.[4] The aim is to uphold the essence of these traditional practices while simultaneously showcasing that expertise and skills in humanities will not become obsolete. It is foreseeable, however, that more interdisciplinary work in larger teams will come to the fore.

The efforts presented as this master thesis are firmly embedded within the e-Ideas project. As already stated, the present project pursues the methodological goal of demonstrating the viability of harnessing computational power for philosophical research. In this pursuit, the author could count on the inspiration, guidance, and concrete technical support from e-Idea's members.[5] This is especially true for the philosophical test case which underlies the validation and illustration of my method in this thesis. I conducted this philosophical research together with the philosophy experts Pauline van Wierst and Arianna Betti who are both working on e-Ideas.

1.2 A First Sketch of the Method

In order to get support from computational systems for any task, one first needs to understand what the task consists in. Then one can instruct a computer precisely what there is to do. In fact, one needs to understand the task so thoroughly that one can explicitly describe all relevant aspects of it. In light of this requirement, I think it conducive to view the task of philosophical interpretation in analogy to doing experimental science: The scientist forms a hypothesis about the nature of a phenomenon in reality; she makes explicit what observable consequences that hypothesis should have in reality, and hypotheses-cum-computed consequences are called her scientific theory; she gathers data about the phenomenon; she confirms or rejects her hypothesis depending on the correspondence between data and her theory, particularly her hypothesis' consequences.

[4]Traditionally, philosophical research involves individuals who close-read and analyze relevant texts by hand.

[5]In particular I am grateful for the support of Arianna Betti, Veruska Zamborlini, Pauline van Wierst, Anna Bellomo, Hein van den Berg, Jelke Bloem, Yvette Oortwijn, Lisa Dondorp, Max van den Broek, and Frank van Harmelen.

Analogously to this, I view the philosophical interpreter to proceed thus: The interpreter forms a hypothesis about the conceptual relations contained in a textual corpus of primary philosophical writing; she states, usually in ordinary language, what the hypothesis is and what it entails with respect to the philosopher's conceptual universe; she goes to the text and points to parts of it that are relevant to the theory about this piece of philosophy; based on the correspondence between text parts and theory, the interpreter rejects are corroborates the theory.

Now, within this thesis I do not assume the role of interpreters myself, but merely build supportive systems for the task. I do so in the following way, under the assumption that the procedure is generalizable: At a first stage an interpreter reads a philosophical theory, as stated in natural language, and comes up with an informal understanding of it. Either alone or with the support of a knowledge engineer she then makes her interpretation explicit by formalizing it in a suitable formal language. This yields a formal correlate or axiomatization of her interpretation.

At a second stage, the axiomatization is encoded so that a computational device can process it. A piece of software can then reveal information about the axiomatization, and indirectly about the interpretation because it is a formalization *of* the interpretation. To provide a concrete example, a computer can check whether an axiomatization is self-contradictory — a logical formalization of some theory turns out to be logically inconsistent — or it can list consequences that follow from it. Both of these tasks can be so tedious[6] to be virtually impossible to realize by hand and the naked eye, thus the computer's contribution discloses important facts to the researcher.

Finally, the computational device's output is linked back to the original philosophical theory. For, ultimately, the researcher is not interested in the formalization per se, but in the light it sheds on the philosophical theory and her interpretation thereof. Here again there are computational ways to support the researcher. The field of computational linguistics offers many techniques and tools to navigate and analyze textual corpora. Some of these can be fruitfully applied to the task of comparing a formalization with the corpus from which it is derived by a two-fold process of interpretation; first, an informal interpretation of the corpus by a philosophy expert, and second, a formal rendering of the informal interpretation to yield the axiomatization. Upon performing this comparison either the interpretation (or parts of it) or the axiomatization (or parts of it) can be confirmed or revised.

The formal languages I chose for axiomatization in this thesis are description logics (henceforth, DLs). DLs are a family of logics. They are fragments of first-order logic, designed to allow for understanding the computational cost of adding or removing logical constructs. Logical languages typically exhibit a trade-off between expressivity — capturing complicated conceptual relations — and computational cost — the time or memory space it takes to perform a computational task; the more expressive a logical language, the more costly the reasoning will be. In effect, DL enable us to balance expressivity with computational efficiency.[5, Chapter 1] That is, respectively, the expected time or space resources to answer a given question and the possibility of expressing a certain concept or relation with our formalism at all. However, expressivity

[6] For any area of interest that is not very small in size and complexity.

will always be limited to a subset of first order logics to guarantee decidability (i.e. there well be an answer, even if it takes long).

There is, to the best of my knowledge, no work, either in philosophy or in the humanities more broadly, that attempts to make use of computational tools in the way I do it here. Thus, my work presents completely unchartered territory where basic principles must first be established. It is, consequently, a rather experimental endeavor. See it as a first pencil stroke on an empty canvas, sketching shapes of possibilities, more explorative than definitive.

It is in this explorative fashion that the thesis derives illustrations and validation from what we are doing within e-Ideas. It is not my primary aim to pursue formalization and further computer-supported processing of the philosophy studied in the project. Rather it should be seen as an exploration of novel methods which have been made possible through technological and informational progress.

1.3 Thesis Outline

The thesis is structured as follows: In Section 2 I give an overview of three kinds of work, each of which is related to this thesis project in a different way. Since my work is innovative insofar as it combines already existing ideas and techniques drawn from approaches relying on formal ontologies, formalization-cum-automated-reasoning, and NLP, it is insightful to consider how these ideas and techniques have been used in the humanities independently from each other. Section 3 presents the method I developed for supporting researchers working on philosophical exegesis. In addition to a schematic description, I here discuss ways in which the method could be concretely realized. Furthermore, I provide some reflection on more foundational, philosophical issues surrounding its methodology. After abstractly describing the method in Section 3, I test it in concrete cases of application in Section 4. Section 5 recalls the contents of this thesis, describes possible avenues of future research as well as general thoughts on the development of computational approaches to philosophical research, and concludes.

So as not to clutter exposition I have chosen to write the thesis assuming the reader's acquaintance with basic definitions and terminology of description logics. For those unfamiliar with the formalism, Appendix A contains an introduction to the basic ideas behind it as well as formal definitions of syntax and semantics.

2 Related Work

The work presented in this thesis relates to a number of very dissimilar projects which operate on the intersection of the social sciences, formal, and computational methods. To structure my attempt at giving an overview, I group the related efforts into three categories:

(i) Research constructing logic-based ontologies for knowledge representation in the humanities.

(ii) Projects using formal means to model theories within the social science for purposes of clarification, analysis, and computational processing.

(iii) Philosophers who employ tools from natural language processing, such as word embeddings, for example, to analyze corpora of text.

While efforts in the second of these groups have a long history, the other two are relatively new endeavours. According to the best of my knowledge, only few efforts exist in these directions. Hence, these fields are presumably still undergoing an experimental and developmental stage. Both for the philosophical use of (i) formal knowledge representation and (iii) tools from computational linguistics well-functioning and standardized methodologies have yet to be constructed, tested, and disseminated.

Techniques pertaining to ontologies and NLP are only being employed in philosophy with relative recency, but they are already widely used in other domains. Therefore, most of the efforts subsumed in categories (i) and (iii) are not so much foundational, in the sense that they aim at developing representation formalisms and computational tools from scratch. Rather they draw on and tweak what has been proven to work in other domains.

In what follows I shall in turn give overviews of work in each of the three categories. My aim is to highlight aspects that are potentially valuable to us and to place my own efforts in an emerging field.

2.1 Ontologies in the Humanities

There is a vast literature on formal ontology building and their use in general. Rather than try to review seminal work of this field, I want to focus particularly on efforts to employ formal ontologies in the humanities. A number of researchers are working on projects whose basic aim is to collect and systematize knowledge about a certain domain of interest within humanities research. Formal ontologies are a way of representing what experts know about a subject in a way so structured as to lend itself to navigation, further processing, interchange, comparison, and educational purposes.[51]

In [6], for one, Bartalesi and Meghini describe the process of building an ontology to support studying the work of the poet Dante Alighieri. In the knowledge base[7] they build, Bartalesi and Meghine capture scholarly knowledge collected from scattered commentaries and literary sources on Dante's primary work. Additionally, they devised an online application to support scholars in navigating Dante's work by querying the ontology.

[7]Roughly put, a *knowledge base* or *formal ontology* is a formal representation of a particular part or aspect of the world. For a more rigorous definition, see Appendix A.

Another example, more geared towards education, can be found in [53]. Here Pasin and Motta present an ontology to navigate one specific philosophical work, Ludwig Wittgenstein's *Tractatus Logico-Philosophicus.* The process of formalization is entirely manual. They develop a prototype tool, *PhiloSurfical*, allowing students to explore the Tractatus from different perspectives, such as historical, theoretical or geographical. The authors express not only an intention to capture Wittgenstein's philosophy, but suggest that their approach to the representation of entities and relations could be used to model the domain of philosophy in general.

In this respect Grenon and Smith, in [30], are very similar. They present a draft of an ontology of philosophy designed to ease navigation of the field in view of its explosive growth as an academic discipline. Concomitantly, they provide a methodology for building ontologies which they claim should be applicable to other domains as well. They explicitly do not want to populate their formal ontology via semi-automatic application of NLP techniques, just like Pasin and Motta theirs is a also a manual endeavour.

Another approach which shares this goal of eventually capturing larger swathes of philosophy is [49]. Niepert, Buckner, and Allen provide the Stanford Encyclopedia of Philosophy (SEP), an authoritative online reference point for philosophers, with a layer of metadata — machine-readable descriptions of the primary content — in the form of a formal ontology. In contrast to Pasin and Motta as well as Grenon and Smith, however, they go beyond hand-formalizing and construct their ontology with a number of different tools. While there is a place for formalizing by hand, they also provide methods to extract information relevant to the ontology in automated fashion. Primary content on SEP undergoes a periodical updating process, therefore the metadata needs to be refreshed each time there is new primary content so it remains current. The authors note that ordinary statistics-based models of linguistic meaning, such as distributional semantics[8] , are ill-equipped for the subtleties of philosophy. Thus, they enhance the updating process with regular queries to domain experts. The ontology is then adapted by combining the experts' machine-readable expertise with automatically collected bottom-up information (based on term co-occurrence statistics of keywords). What is noteworthy about Niepert, Buckner, and Allen's effort is their readiness to admit limitations of formal and computational tools. They state that as of right now, if not in principle, the practice of philosophy cannot be fully automated. Hence, the systematic and repeated use of interaction with actual human experts must form part of any successful method.

In light of these insights, Buckner, Allen, and Niepert follow up their 2007 paper in [16] with a more general reflection on the project of elaborating a formal representation of the academic discipline of philosophy. The challenge they identify and tackle here is the numeric explosion of academic publications, researchers, and institutions in recent years which makes it hard to enter the field, navigate it, and distinguish the irrelevant from the pertinent. To bring structure to the unruliness, the authors propose to use dynamic computational ontologies which are constructed in tandem by domain experts and automated techniques. They illustrate what they have in mind by adding to their previous attempt of developing an explicit, formal, and computationally-tractable representation

[8]See Section 2.2 as well as Section 3.2.3 for brief explanations of the ideas behind *distributional semantics.*

of SEP's contents. To meet this goal, authors submitting or updating articles to SEP are solicited for their expert knowledge. They are asked about taxonomic relationships and term relatedness within their submission. Additionally, experts confirm or falsify unvalidated information gathered automatically via NLP techniques. Only the information validated through the expert-loop is then used for the actual provision of metadata on SEP. Although Buckner et al. do not elaborate much on this, their quality-control loop foresees experts checking novel inferences gathered via automated reasoning algorithms. Hence, such an approach could even point researchers to new findings, thus automating parts of the creative work involved in research. With this sidenote they are, to my knowledge, the only authors in the formal ontology camp in philosophy who are considering to harness the computational powers of their systems to obtain new insights, instead of simply managing what is already known.

2.2 Computational Linguistics for Philosophy

Due to the awe-inspiring size of their dataset and the abundant avenues of research their work points to, Michel et al.'s [46] stands out among the relevant papers in this group, although they have not addressed any philosophical question in depth. They report having collected an impressively large dataset of digitalized books, with their corpus covering roughly 4% of all books ever published. Using this vast corpus the authors use a quantitative approach to the study of changes and trends in human culture as reflected in language. In this particular paper they take cursory looks at a diverse set of issues in linguistics and history.

[38], on the other hand, presents a much more focused investigation. The computational linguist Herbelot collaborates with the philosophers Redecker and Müller to apply techniques from distributional semantics to analyze concepts relating to gender and race in large text corpora. The authors show that collaborations between computational linguistics and the humanities are possible, provided that the computational methods are neatly aligned with the research questions. In their paper, the formal representation of meaning from the computational side indeed matches the concept of meaning underlying the research question in feminist philosophy, to wit, how certain concepts are used in linguistic discourse. For example, the authors claim that the use of gender concepts in discourse, eg. *woman*, is influenced by and influences how gender roles are perceived and formed.

The authors are looking for hard evidence to support their philosophical claims. They are looking to avoid two common points of critique raised against attempts to perform discourse analysis in traditional, non-computational ways which means close-reading and analyzing the use of relevant concepts in a small selection of pertinent texts. Critique number one goes that the selection of the text could be biased, and other texts would not warrant the same finding; number two, the selection of analyzed texts is too small, hence the finding cannot be expected to be prevalent elsewhere.

By automating the process of analyzing the meaning of concepts on the basis of a very large set of discourses — English language Wikipedia in this case — the authors try to counter both objections at once. Within the distributional semantics approach they take, the representation of a concept's meaning relies on the assumption that this meaning is determined by the context in which it appears. Because a computer can very quickly process all the occurrences and contexts of a concept within a large amount of data, the authors obtain formal representations of a concepts's meaning and assess the (formally captured) semantic relations it is involved in with other concepts.

An important observation for my own purposes is that Herbelot, Redecker, and Müller's work is convincing because the research questions they address naturally fits their choice of formal and computational methods. It thus illustrate nicely that such happy fusions are possible, but tools and questions need to be matched carefully. Stubbornly forcing a formalism on a problem is to set up a failure.

In [52], Overton chooses a computationally simpler approach than Herbelot et al. He reports using simple statistical methods on a dataset of 781 articles from the journal *Science* to investigate the concept of *explanation* in science. The author uses two methods: First, he compares the relative frequency with which words that are indicative of a concept's relevance are used in the corpus. Second, from all the sentences in which such a word occurs he randomly samples a small selection. He then classifies the type of explanation present in the selected sentences by hand, thus obtaining example cases which are immune against charges of biased selection. Aside from Overton's philosophical conclusions (*explanation* is indeed an important, yet diverse, part of science), he assesses his experimental method positively. For one, firmly establishing some facts of the matter quickly and efficiently has made his research easier; also, his results are more likely to be methodologically robust because they are not made on the basis of hand-selected cases but on cases chosen via random sampling.

In the forthcoming [13], Betti, van den Berg, Oortwijn, and Treijtel take a similar approach as Overton. These authors seek the likely origin of the concept *conceptual scheme* in W.V.O. Quine's philosophy. They use NLP to look for the bigram "conceptual scheme" within a corpus of articles from scientific journals around the time when the concept first came up. They collect metadata for each article (author, institution, year, etc.) and hand-annotate the passage where the bigram occurs. Using these data the authors apply descriptive statistics to support their claims about the likely diffusion of the concept. Just as Overton does, these authors stress the message that such techniques are meant to support philosophical research and not to supplant it in any way.

A last important piece of work is described in [64] where van Wierst, Vrijenhoek, Schlobach, and Betti report on the project Phil@Scale which is directed towards the development of computational methods to support researchers working in philosophy, particularly its historical aspects. With Phil@Scale the authors seek to tackle two related issues they see in traditional methods of philosophical research, thereby echoing Herbelot et al.: One, coming up with and testing hypotheses by individually close-reading relevant texts is a very slow and arduous process. Two, the fact that individual researchers pursue their analyses in isolation, and often without fully disclosing their standards of evaluation, poses the risk of reporting biased results. To counter these two issues, the authors built the tool *SalVe* to support researchers in a quantitative way. *SalVe*, too, should not replace traditional methods, but complement them by, for example, determining which parts of the textual material are most pertinent for answering a given research question. SalVe takes a digitalized corpus, suitably divided into parts, as input data and enables researchers to do advanced word searches, determine similarity relations between the corpus' parts, display the co-occurrence of words, and indicate words that seem most relevant for particular parts of the corpus. For these computational facilities the tool relies on the well-known technique of word counts. The authors describe the technical aspects of *SalVe* and apply it to a concrete philosophical research question. They conclude that its use has indeed significantly sped up the research process and, at the same time, made it more thorough.

It becomes visible from these few early projects that statistics-based NLP methods can be fruitfully applied to philosophy, but that these methods should currently be thought of as supportive tools. Even in the case of Herbelot et al., where the research question is extremely neatly aligned with a rather sophisticated computational tool, human researchers must interpret its output and conduct careful analysis.

2.3 Theory Building with Formal Tools

The earliest example of work using formalizations to assess and strengthen scientific theories I found is [54] where Peli, Masuch, Bruggeman and Ó Nualláin employ first-order logic to formalize a particular scientific theory, *Organizational Ecology*, from the field of management science. Upon axiomatizing the theory, the authors manually compute theorems implied by the axiomatization. In light of these implications they evaluate Organizational Ecology. Peli et al.'s idea to formalize a scientific theory in order to compute its consequences is involved in my method as well. Differently from them, however, I did not compute the consequences of axiomatizations by hand as they do.

In this sense, [41] and [40] are two efforts that bear more resemblance to my project than Peli et al.'s. In [41] Kamps and Masuch seek to improve on Peli et al. by automating the process of computing partial deductive closures of Organizational Ecology's axiomatization. The authors propose a cyclical process as illustrated in Figure 1. They view theories as propositional systems which are logically structured. Formally speaking, a theory is a set of statements closed under the inference rules of the logic one adopts. The authors make a difference, however, between what a theoretician explicitly states (what the authors call a *premise set*), the complete, deductively-closed theory (*complete theory*), and theories in between, made up of premise set plus some subset of the complete theory (*intermediate theory*). In addition, the authors call *theoretical expectations* whatever an expert knows about the domain of interest, regardless of whether it is represented formally, in ordinary language or otherwise. In analogy to the cyclical process of empirical research, consisting of preliminary identification of the issue at hand, the formulation of hypotheses, the gathering of data, and hypothesis testing, the authors then describe a logical cycle capturing the interaction between theoretical expectation, formalization, and the computation of consequences.

Theoretical expectations are formalized, yielding an intermediate theory. A basic test for this theory is whether it is logically consistent. Inconsistency should lead to either a revision of the formalization or of the theoretical expectations. Subsequently, the (consistent) intermediate theory's consequences are computed to obtain a partial closure. Whatever new conclusions are thus discovered can either a) confirm the theoretical expectations, so they are strengthened, or b) contradict the theoretical expectations, so they will have to be revised, or c) transcend the theoretical expectations in the sense that they present unexpected, but compatible, insights with which the theoretical expectations will have to be updated.

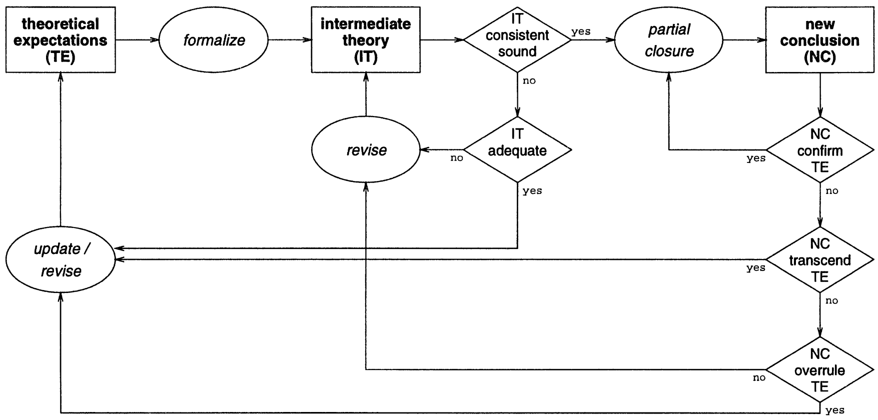

Figure 1: The Logical Cycle in Kamps and Masuch [41]

In a 1999 paper, [40], Kamps follows up the logical formalization-cum-automated reasoning approach to Organizational Ecology by considering its application to social sciences in general. Specifically, the author here aims to unite insights from the fields of logic and philosophy of science to establish evaluation criteria for axiomatized theories in the social sciences. The criteria Kamps identifies as important are tailored so as to be tested by using computational support from automated theorem provers and model searchers. However, due to the undecidability of first-order logic there are systematic limitations to this approach. The author thus urges readers to view the criteria's test not as a rigid, final result, but rather as useful feedback to be incorporated during the formalization process. According to Kamp, axiomatization is best not viewed as the final step in the lifetime of a scientific theory where its final version is formally enshrined. Instead it should play a dynamic role in its development.

There is, of course, a whole host of work seeking to formalize scientific theories without much intention to process the axiomatizations automatically. Often this is done for the purpose of clarity. A perfect example from the field of philosophy is [2] where Achourioti and van Lambalgen reconstruct aspects of Immanuel Kant's logic in order to shed some light on the concepts and relations therein, while at the same time showing the consistency and coherence of Kant's system. Such efforts relate to this thesis project because formalizing philosophy in order to process it forms part of my comprehensive computational method. My proposed method is presented in Section 3.

3 The Method: Formalization, Automated Reasoning, and Natural Language Processing for Philosophical Interpretation

In this section I present and discuss a method to support philosophical researchers through a combination of formalization, automated reasoning, and NLP techniques.

For the formalization I use DLs, a family of logics combining expressivity with good computational properties. They are developed particularly to model and process human knowledge in a variety of domains, and thus there exists a wide range of techniques as well as ready-made software packages for researchers in philosophy to access and reuse.[31] Knowledge formalized in DL can be automatically processed by a computer in ways that no human researcher plausibly could.

NLP is a field of research that uses computational tools to build natural language text or speech applications. It is related to computational linguistics, an area of linguistics research relying on computational techniques to advance scientific understanding of natural language phenomena.[19] The knowledge thus obtained can be used to develop computer systems serving a wide variety of purposes. Principal applications include translation between natural languages, artificial intelligence, user interfaces, and summarization of text or speech recognition.[19] Importantly for my purposes, methods from NLP can be used to navigate and analyse large bodies of machine-readable text, called *corpora*.

As indicated in the previous section on related work, I am not aware of any work in philosophy that focused on combing computational techniques from NLP with DL-based knowledge representation and, crucially, automated reasoning. Also, some recent efforts focus on NLP for philosophical research, but this has not been combined with logical methods. In the specific way in which I make use of these tools my work is thus unprecedented. Although there are DL-based ontologies to provide metadata for philosophical corpora, for navigation or didactic purposes, automated reasoning is at most peripheral to these projects. For other research projects automated reasoning lies at the core, but their formalisms greatly differ from my DL approach. I now describe my method which draws on these fields in very broad terms at first. Afterwards, I shall discuss each of its aspect in some more detail.

3.1 A Protocol to Test Philosophical Interpretations

All too often discussions concerning the methodology underlying philosophical interpretation are unsatisfactory. Interpreters are not explicit about crucial aspects of the method they use when coming up with an interpretation of a primary text. It is left unclear what would count as evidence for a particular interpretation, and even less clear what principles (should) underlie the judgments involved in weighing up conflicting evidence. Typically, the support for a given interpretation consists in pointing to a selection of passages from the primary text, but little guarantee is offered that the strength of these selected passages is not undermined by the interpreter's bias. After all, she could have come up with a particular sample of passages to support an otherwise untenable view.

An issue related to the problem of bias is that sometimes an interpreter's preliminary notions with which she approaches the text for close-reading are left opaque. But, as no text carries its interpretation on its sleeve, an explicit account of an interpreter's conceptual scheme would be needed to assess a particular attempt at interpretation. This omission can arise because of the vagueness and ambiguity of ordinary language in which the interpretation is presented, or simply because of lacking scientific rigor. It is sometimes left under- or entirely unspecified what concrete hypotheses are under scrutiny in exegesis. In addition to vagueness and lack of transparency there is the mundane possibility of human error. Philosophical writings can be long and exceedingly difficult, both linguistically and conceptually. Human attention is limited and so is the time to look for the parts of primary texts that really matter for a particular research question.

A principled answer to some of the issues lately sketched is to treat philosophical interpretations as scientific hypotheses about a *phenomenon* within the primary text. These hypotheses are corroborated or rejected by the *reality* of the primary text. When subscribing to this plan, hypotheses must be stated explicitly so that their consequences can be computed. The hypotheses plus these consequences following from them build a theory which the interpreter claims to be a correct description of the primary text's content. This theory must then pass the test of comparison with the textual corpus. In case the corpus refutes parts of the theory, the theory must be revised.[9]

I use computational systems and techniques to support philosophical interpretation on these principles. A general and high-level description of a way to apply theses computational methods is represented in the diagram in Figure 2. Arrows indicate in what sequence the method's steps are carried out. Boxes stand for representations or data which undergo transformations or processes depicted by ovals. Rhombi indicate decision nodes where the satisfaction of a condition decides on how to proceed. Finally, there is one *or*-node, represented in a circle, where there are two possible ways to proceed, but no clear condition is specified.

[9]Unless, of course, the philosophical theory expressed in the corpus is taken to be inconsistent. It is probably fair to say, however, that the principle of charity is a very forceful requirement for any project of exegesis.[21] Hence, the presumption of consistency is essential and not light-heartedly abandoned.

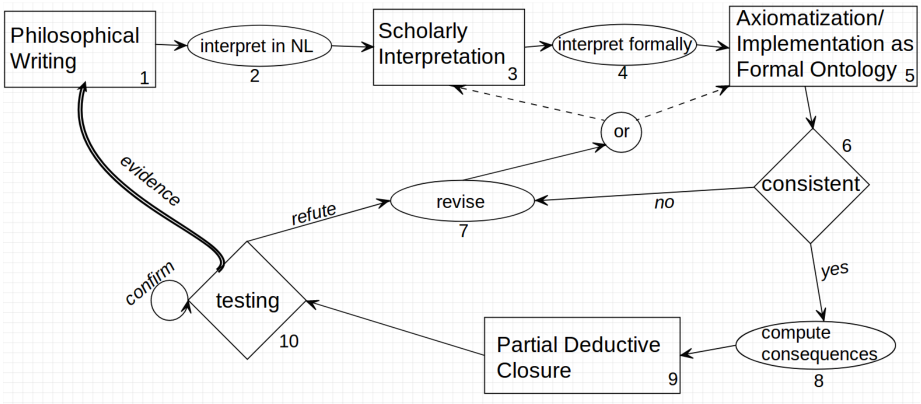

Figure 2: The Method in a Diagram

Initially, there is a primary philosophical text we would like to interpret. A human expert might very carefully read the text in the traditional manner and provide us with an interpretation thereof. This interpretation typically takes the form of a secondary text in a natural language, like English or Kiswahili. Possibly, the expert already takes a step towards formalization and expresses her interpretation in semi-formal style, in preparation of what follows. In a next step, the informal or semi-formal interpretation is fully formalized in DL to a) make it completely unambiguous and explicit about what is expressed, and b) give the interpretation a form which a computer can process. The formalization is a set of formal expressions, also known as *axiomatization*, where each expression is one axiom. Subsequently, the axiomatization is implemented in a DL editor which is a piece of software to work with formal ontologies. Upon its implementation as a formal ontology we can apply automated theorem proving software to our axiomatization. This automated reasoner computes a part of the deductive closure of our axiomatization — the set of all derivable consequences — so that we get an overview over what our formalized interpretation of the primary text entails. Obtaining the partial deductive closure allows us to test immediately whether our axiomatization is consistent. Checking an axiomatization's consistency is a standard facility of DL reasoning software. In case of inconsistency, some reasoners even provide explanations of where things went wrong. In case it is consistent, however, we treat the partial deductive closure as the fully specified formal theory of our interpretation. We then test this theory against the primary text. We assess whether the theory, consisting of axiomatization plus inferred statements, can be justified as a plausible interpretation of the primary text. What we are looking for is evidence in favor or against the theory's truth. We try to gather this evidence by a series of different means, ranging from traditional expert assessment to automated and highly sophisticated NLP techniques. Because the data against which our exegetical theory is tested consists of text, techniques from computational linguistics are uniquely suited to support the search for evidence of this kind. To end the cycle, we either reject or accept (parts) of the theory. In case of rejection we go back to either informal or formal interpretation and revise our theory.

In what follows, these steps are articulated and discussed in some more detail

3.1.1 Axiomatization

The diagram in Figure 2 depicts the outset of my method as the box number 1, representing an uninterpreted piece of philosophical writing, and the circle number 2, standing for the initial act of informal interpretation. This process yields an informally stated scholarly interpretation of the philosophical writing, depicted in box number 3. Based on this informal interpretation, the process of formal interpretation is applied — oval number 4 — which results in an axiomatization of the informal interpretation. This formal correlate to the scholarly interpretation is presented as box number 5.

Let it be clear that finding an adequate formal correlate to a natural language theory is no easy task and always subject to heavy philosophical assumptions.[60][50, Chapter 3.3] It is important to note that formalization, just as the informal interpretation, always involves an act of interpretation.[55, Chapter 2] Different interpreters might arrive at different formalizations of the same informal interpretation, due to the context-dependency and ambiguity of natural language. It is doubtful in principle whether a formalization in a given language can be thought of as a process of mapping, in the sense of a mathematical function, a piece of text in natural language to the one correct corresponding set of expressions in the formal language.[10][7]

As an alternative to a full procedure mapping natural language text to a formalism of choice, one could rely on criteria or guidelines for the production of adequate formalizations.[11] One of the more recent, and perhaps the most thorough, work offering such criteria is Georg Brun's book *Die richtige Formel*, [15].[7] [12] Aside from these works, relatively little sustained analysis has been given to standards of correctness of formalizing natural language.[50, p. 97]

While there are few authorative reference points for the general project of formalizing natural language text, my interests are slightly different. I pursue the specific purpose of using DLs as the basis of formal ontologies — symbol structures modeling some aspect of the world while abstracting away from irrelevant detail.[5, Chapter 10] Due to the wide usage of DLs for modeling some universe of discourse, a number of guidelines are available, stating basic principles of ontology modeling. Among these stand out [5, Chapter 10], [31], and [51].

Note that the informal and the formal interpretation step need not necessarily be carried out by the same individual. One clear benefit of a separation of tasks is that whoever is in charge of the formalization process does not need a great deal of philosophical expert knowledge herself, as her role is restricted to that of a supportive knowledge engineer. Most likely, however, it would come down to a collaboration between experts in philosophy and formalization.[13]

---

[10] Although some proposals exist, think of Davidson's programmatic statements that such a procedure is a necessary precondition for a theory of meaning or the more hands-on procedures of Chomsky and Montague, there is currently no generally applicable and effective formalization procedure available.[7]

[11] The literature mentions only few such studies. Among them are [58],[22],[23], and [14].

[12] See also [7] itself for a take on necessary and sufficient conditions of adequacy of formalizations.

[13] Nonetheless, for the case of DL-based languages there are efforts to systematize the art of ontological modeling. In [33] and the less technical [32] Guarino and Welty apply insights from analytical metaphysics to find a more scientific basing for ontology engineering. The authors have developed a methodology, *OntoClean*, to help engineers in avoiding modeling pitfalls.

One problem one could run into as we formalize is that our DL formalism of choice is not expressive enough to capture the philosophical notions involved in an interpretation. As previously stated, DL are deliberately limited languages so as to grant good computational properties. It seems possible, however, that there are systematic ways to compromise on some of these issues in particular cases. An example of such a way can be found in [34] where Guizzardi and Zamborlini outline a strategy to partially map axiomatizations in a more expressive language (OntoUML) to the less expressive web ontology language (OWL).

3.1.2 Implementation as an OWL/DL Ontology

Once an adequate formalization is in place, it is implemented in an ontology editor. Due to the use of OWL/DL ontology in a great number of domains of application, especially the semantic web[14], there is a sizeable and growing number of ontology editors and reasoners available. Some outstanding software is free and open-source, notably Protégé, a flexible and widely used ontology editor. Perhaps due to its broad user-base, there is excellent documentation available for Protégé. Other editors, such as OntoStudio and TopBraid Composer, require licenses but come equipped with visualization tools. For a detailed comparison of a few ontology editors from the semantic web context, see [3]. In this article the author states that:

> It is quite clear that Ontology development is mainly an ad-hoc approach. Among several viable alternatives, a user needs to find which one would work better for the projected task and which one easily and effectively can be maintained and expressed.

Thus, it appears that the choice of editor boils down to pragmatism, once the question of formalism is settled. Technically speaking, one could design and implement software to build and maintain an ontology that caters especially to the needs of whatever philosophical research one is currently conducting. Such an effort can turn out to be monumental, however. Hence, one might often be much better off using software that is readily available and well-documented and adapt it to one's purpose.

3.1.3 Consistency Requirement

Upon implementing our axiomatization as a formal ontology in a suitable editor, we can check whether the axiomatization is consistent. This check is presented diagrammatically in rhombus number 6. If this requirement should not be fulfilled, we move along the edge labeled with "no" leading from rhombus number 6 to the oval number 7. Three possibilities arise, from which two are represented in the diagram: The first possibility is that the informal interpretation we are working with is true to the primary text, and the axiomatization truly captures the informal interpretation, but the philosophy we are interpreting was logically inconsistent all along. In such a case, any effort at systematic reconstruction is futile and we best abandon the attempt to arrive at a consistent interpretation,

[14]The semantic web is an initiative to add a layer of conceptual information to content on the world wide web in order to overcome the glaring absence of structure, and the consequent limitations to navigation and processing based on surface form only. This additional semantic layer is represented by formal ontologies.[5, Chapter 14]

for such a reading would directly and clearly go against the primary philosopher's intentions. Because it contravenes the basic assumptions of my method, this option is not represented in the diagram.

There are, however, a second and third possibility consisting in either faulty informal interpretation or axiomatization. Either we are working with an erroneous informal interpretation which has been axiomatized correctly and consequently leads to an inconsistent set of formulae; or our informal interpretation was in order, but our axiomatization thereof has introduced the inconsistency. When presented with logical inconsistency, we have no way of seeing where the error has occurred. It could even have occurred at two levels of interpretation at the same time.[15] In any case, if a reasoner detects that our formal interpretation is inconsistent, we need to reconsider both levels of our interpretation. Thus, there is an "or"-node in the diagram indicating to revise either box number 3 or 5, or both.

One way to systematically check only the formal interpretation would be to implement axiomatization step-wise, adding one axiom at a time while always checking the current state of the logical closure. This procedure presupposes, however, that the interaction of the axioms is fairly transparent. In case it is unclear how the axioms come to imply theorems, one could also compute all possible combinations of axioms and run the reasoner for each to analyse under which setting the problems occur. The problem, of course, is that the number of possible combinations grows exponentially in function of the number of axioms. Therefore, this approach only works for rather small axiomatizations.

3.1.4 Computing Consequences

If the axiomatization passes the consistency check, we move along the "yes"-edge from rhombus number 6 to oval number 8 where we consider the editor's reasoning output. Here also, a pragmatic stance should guide our choice of reasoning software. Many of the editors available come with built in theorem provers.[1][5, Chapter 8]

However, particular logical languages or reasoning tasks might require tailor-made reasoning software. DL reasoning and its optimization is a lively area of research, so novel reasoning algorithms might suddenly render certain tasks feasible.[20]

Once we have settled on a suitable reasoner, we obtain the partial deductive closure of our axiomatization from the reasoning software. This corresponds to box number 9 in the diagram. Loosely speaking, the deductive closure of a set of formulae in a logic is the set of all formulae which can be deduced from the original set. The output is only a *partial* deductive closure because any tautology forms part of the complete closure, since tautologies follow from anything. Yet, we are usually not interested in tautologies, nor can we explicitly represent an infinite number of them. In this sense the reasoner's output is only a selection of consequences.

[15]Wandering even more into the realm of possibilities, we cannot even strictly trust a consistent axiomatization. It is conceivable that a consistent axiomatization reflects a *false positive* in the sense that it falsely suggests that the philosophy is logically consistent and our two levels of interpretation were carried out dutifully. Such an error can occur if two wrongs made a right, when, for example, a faulty informal interpretation was poorly axiomatized so as to result in a consistent formal interpretation.

To be formally precise, we could follow [40] and let Σ denote the set of expressions in our axiomatization. A formula φ, then, is a theorem of that axiomatization if and only if it is a logical consequence of the axiomatization, that is, if $\Sigma \models \varphi$. Finally, the full formal theory — $\mathcal{T}$ — is the logical closure of our axiomatization, i.e. the set of all theorems: $\mathcal{T} = \{\varphi \mid \Sigma \models \varphi\}$. The partial deductive closure — call it $\mathcal{P}$ — forms a subset of $\mathcal{T}$ because of the aforementioned reasons.

3.1.5 Testing

If inconsistency is not an issue, then what we have in hands is a formal theory of our interpretation.

We can now assess whether the statements in $\mathcal{T}$ can be justified as plausible interpretations of the primary text by going back to the corpus. What the automated reasoning step has brought us, apart from the basic test of consistency, are more possibilities to assess the adequacy of our interpretation, since the partial deductive closure is usually a superset of the axiomatization. That is to say that usually $\Sigma \subset \mathcal{T}$.

The testing phase is represented by rhombus number 10 in the diagram. It is presented as a decision node, since finding contrary evidence should lead us to revise our interpretations, as indicated by the edge labeled "refute" going to oval number 7. In case our testing strengthens the partial deductive closure, we follow the reflexive "confirm"-edge that loops back into the testing phase and continue testing the theory. The double line between rhombus number 10 and box number 1, labeled with "evidence", indicates that the testing phase involves going back to the original piece of philosophical writing.

The guiding question of the testing phase is: What kind of evidence can be found in the primary text for the statements in the partial deductive closure? To discover and collect this evidence we rely on a series of different means, from asking philosophical experts to the employment of automated NLP tools, and mixtures of the two. I now present a list of tools that we use (or think about using in the future) in the e-Ideas project. Of course, this list is not meant to be exhaustive as I merely intend to illustrate what kind of possibilities are around.

It must be said as a preliminary note, however, that e-Ideas is in possession of a fairly clean corpus of philosophical writings which researchers in the group are interested in. Building such a large and clean corpus has been a very laborious effort.[8] The process involves getting a hold of high-quality scans of the relevant works, finding or building reliable OCR[16], and a considerable amount of post-correction to eliminate errors introduced in either of the first two steps. Although it is likely that this process will become easier in the future, as more books are digitalized and freely available and as the performance of the necessary software improves. Nonetheless, it should not be underestimated how hard it is to get hold of a high-quality corpus.

With such a corpus in hand, however, the comparison-step, consisting of linking the consequences back to the data of the corpus, is supported in a number of computational ways. These tools could be ordered by degree of computational involvement.

[16] Optical character recognition software turns digital images to machine-readable representations of text.

One entirely non-computational way is traditional: Ask an expert to access her memory and knowledge of a philosopher's writings in order to manually navigate the corpus and find refuting or corroborating evidence for a particular statement in the theory. A variant of this is probably needed for all tools I am about to present. Human expert knowledge is unlikely to become obsolete within the foreseeable future, therefore experts will certainly need to be kept in the loop.

Perhaps the smallest computational step would be to use a simple search function to navigate the corpus. This would allow one to find passages within a corpus where a given string occurs. A somewhat simple computational tool is *SalVe*.[17] It was tailor-made for work along the lines of e-Ideas and has been proven to speed up the research process while also increasing its thoroughness.[8]

A much more sophisticated computational tool is *Ariadne*. The technology underpinning it are *word embeddings* in which words are semantically represented — embedded — in a vector space. Words that are similar in meaning are mapped to points in each other's proximity in the vector space. Ariadne features a visual interface allowing users to explore the entities in this space and thus to interactively navigate the context of query terms in a textual corpus.[62] Because of its visual interface, no great technical knowledge is required to perform the queries.[8]

Ariadne was originally developed for a corpus of scientific articles, but it has since been successfully applied to other types of corpora[18], specifically for philosophical data.[8]

It is an online tool, so the speed with which it answers queries was a big concern in its development.[42] Therefore, the tool now requires an off-line preparation phase where each entity is semantically represented not based on the co-occurrences in the data, but on the context it shares with other entities.[42] Then the online interface suitably reduces the number of hits to a query so that they are easy to grasp and navigate.[42] When a user enters (possibly multiple) search query terms, these are matched to entities in the semantic space.[42]

According to a comparison between different embedding techniques, carried out by the designers of Ariadne, their tool is particularly fit for the task of information retrieval: Given a search term, it indicates what parts of the corpus fit the query best.[42] This information retrieval facility can be very powerful. In practical application Ariadne enables a human researcher to detect passages in the primary text that are about given concepts.[8]

What is needed for this is a list of *labels* assigned to each concept modeled in the ontology. Labels are natural language expressions of a concept. Because the corpus is processed by Ariadne based on the surface form of Bolzano's writing, there is a need to bridge the conceptual level of the formalization with the level of word-shapes. Once this preliminary work is done, we can navigate the corpus in order to find the most relevant passages for or against a certain item in our partial deductive closure.

Another computational approach consists of using a tool to automatically extract grammatical relations within the corpus, a so-called *dependency parser*.[19] Perhaps one could use the dependency parser to determine the dependency re-

[17] It has already been introduced here 2.2.

[18] The tool uses no language-specific preprocessing, so non-English corpora pose no problem to it.[8]

[19] See [43] for a thorough treatment of theory and practice of dependency parsing.

lations between grammatical constituents within the sentences of a textual corpus. If the extraction is successful, these relationships, in turn, could be used to obtain more information on how terms relate to each other in the text along grammatical lines. For example, if a term is a noun we would like to determine its objects in case it occurs as a subject, and its subject if it is an object. If we know what terms are associated with a particular philosophical concept we are interested in, we could use the output of the dependency parser to see all the relations of these terms. Because the terms are the concept's labels this would allow us to see see if among the relations we find those which are predicted by the partial closure of the formal ontology. The viability of this idea has been discussed in e-Ideas, but we have not taken concrete steps towards realizing and testing it. A possible drawback of this technique is the amount of manual work it requires, since presumably a part of the corpus would have to be carefully manually annotated for the method to work.

A different take on the same computational idea would be to focus not on grammatical but on philosophy-specific relations.[20] An expert philosopher would first have to annotate a portion of the corpus with layers of metadata to identify the relevant relations. One would then train a machine learning classifier to extend the annotation to all of the corpus. This too could turn out to be challenging because of the work it requires. In addition to the high labor requirements, it is unclear what relation would be both philosophically valuable and learnable at the same time.

A final, quite sophisticated computational approach would involve training a distributional semantic model on the philosophical corpus. A standard approach here would be to use a model from the *Word2Vec* family which are based on neural networks.[29] However, Word2vec is unlikely to work in philosophical contexts because much more data is needed to train these models than typically available in philosophy. Perhaps it would be possible to depart from concepts which are already known in order to compare the performance of Word2Vec models, which are based on neural networks, with other count-based models for word embeddings. Hereupon, one could use operations on the models to see whether a given conceptual relation is as the reasoning output of the ontology predicts.

3.1.6 Further Reflection

This concludes the overview over a computational method for philosophical interpretation as I envision it. I deliberately choose a presentation that does not specify completely how the individual steps of the method are to be realized. Rather, I try to show a number of possibilities and limitations at each step, so that researchers interested in a method along these lines may feel inspired to adapt what they can use for their own purposes. I shall shortly present my own attempt concrete realizations of the method when I apply it in Section 3.

Before I get there, however, a further discussion of methodological assumptions is in order. To the best of my knowledge, this type of computational methodology presents completely unchartered territory in the field of philosophy. Given this status of methodological novelty and experiment, two stances could be taken towards my method: Under the first, perhaps more practically-

[20]For such efforts in other fields see [35], [25] or [27].

minded, viewpoint, I decide to go ahead and think about applying the method sooner rather than later. Further methodological considerations makes way for concrete practical application, so we get to see how the method fares in practice. Thinking about philosophical foundations is postponed to a later point in time. Under the second, perhaps more philosophically-minded, point of view, I proceed more cautiously and first analyse thoroughly what I have in hands here before engaging in a host of shaky investigations.

I believe that both of these stances have their respective merit. Therefore I now briefly treat them in turn, starting with the more philosophical considerations before also treating some issues related to practical application.

3.2 Philosophical Perspective

I have stated it before, but it bears repeating: Neither formal, mathematical redescription nor the use of computational tools and systems is likely to replace traditional philosophical analysis any time soon. What I hope to achieve, for now, is to build formal and computational systems that support human researchers in their efforts.

Because simply issuing this caveat is not enough, I take further looks at a number of issues related to philosophy and the use of such methods. The first of these issues concerns the proper role of formalization in philosophy. These considerations are guided by Rota, Pinosio, and Dutilh Novaes who all have written on the relationship between mathematical or formal methods and philosophy.[57][56][50] The second issue concerns the reasoning process which I automate through my use of formal ontologies and theorem proving software. The question I address here is: What guarantee do we have that the inferences enabled by our formal logic and carried out by our reasoning software coincide with the reasoning process in traditional research? A third and final bundle of issues I touch upon is how more sophisticated NLP methods fit in with my philosophical setting. Such sophisticated methods may bring with them very substantial assumptions about philosophically sensitive concepts such as *meaning*, for example.

3.2.1 Formalization in Philosophy

Upon the arrival and popularization of major innovations in formal logic, the 20th century has been marked by the application of logical formalisms to a wide range of philosophical questions. As Gian-Carlo Rota, a prominent mathematician and philosopher, laments, this has sometimes led to abuse of both philosophy and formal tools.[57] He and Riccardo Pinosio, a philosopher working on the formalization of Kant, particularly disapprove of unprincipled attempts to resolve philosophical problems by capturing fundamental informal philosophical concepts in formal axiomatizations to then go off and prove results, supposedly about the informal philosophical concepts. Rota and Pinosio point out that attempts of this sort often break down because the relationship between the informal philosophical notions or theory and the axiomatization is unclear.[57][55, Chapter 2]

We agree that for this philosophical strategy to work the relationship between formal and informal philosophical theory must be carefully established. The formalism should fit or align itself with the informal philosophical theory.

Consider, to give a concrete example, an attempt at formalizing an inconsistent philosophical theory in classical logic. From this axiomatization one then quickly moves on to prove further consequences, supposedly entailed by, but not explicitly mentioned in, the informal philosophical theory. Such an exercise is devoid of point, since anything can be proven from an inconsistent axiomatization in classical logic.[17]

It must be kept in mind that the process of formalization involves an act of interpretation.[7] Moving from an informally stated theory to a precise and largely unambiguous formal theory requires us to decide on particular interpretations among the possibilities left open by vagueness or other sources of underdetermination. When formalizing, we are forced to abstract from aspects of the informal theory, what we deem inessential is left out, and what we deem essential is enshrined in our axiomatization. Examples and other illustrations are left behind, ambiguities and other uncertainties collapse into singular ways of interpretation.[55, Chapter 2]

Think of the exegetical method lately described as involving different levels of interpretation. On the null level there is the primary philosophical text which stands uninterpreted. A scholar's informally stated interpretation of the primary text constitutes a first level of interpretation. An axiomatization of the informal interpretation of the source text stands on a second level of interpretation. The axiomatization's relationship with the primary text is thus only indirect, it is mediated by the informal interpretation of the text. This must be kept in mind, especially when reasoning about the axiomatization. We reason primarily about the axiomatization, secondarily about the informally stated interpretation, and only remotely about the content of the original philosophical text. This being said, however, how close a formalization can be aligned with philosophical text always depends on the specific text in question. There exists, of course, very ambiguous philosophy where many formal routes could be taken, and then there is philosophy which is presented nearly or fully formally already. In the latter case there might be very little reasonable disagreement about adequate formal correlates.[21]

Another important mental slip Rota denounces is to think that imitating mathematics requires us to start out with clear definitions from which we then compute theorems.[57] First of all, in philosophy one usually ends with a definition and does not start with one. Concepts such as *truth* or *mind* are not clear from the start of philosophical inquiry but are clarified over a lengthy process of deliberation. Secondly, Rota stresses that in mathematics the theorems motivate the axioms just as much as the axioms the theorems. Axioms are chosen in light of the theorems provable from them.

> A good definition is "justified" by the theorems one can prove with it, just like the proof of a theorem is "justified" by appealing to a previously given definition. There is, thus, a hidden circularity in formal mathematical exposition.[57, p. 172]

The question then becomes: What is the role of formalization in philosophy if it is not completely solve philosophical problems in an axiomatic system? A possible answer lies in Rota's remark that there is cyclicity in mathematical exposition. Although any axiomatization of a philosophical theory requires

[21] See, for example, [44] where the formal correlate is practically contained in the philosophical paper already.

a preliminary understanding or pre-formed interpretative attitude towards the theory, the process and results of the axiomatization open up new exegetical possibilities which were not previously considered. There is, thus, a place for what Pinosio calls a “virtuous hermeneutic cycle”[55, p. 11] where formalization and non-formal understanding of the philosophical theory update each other iteratively. Under this view, both the formalization and the informal interpretation are perennially unterminated and incomplete, they are only stages in an on-going hermeneutic process. The formalization is best viewed not as an end in itself, but as a tool to advance and refine our understanding of the philosophical text in tandem with our informal theory.

This points to the particular benefit of formalization that, just as in scientific theory building, it requires us to focus on the essential aspects of a phenomenon and abstract from the unimportant. Moreover and very importantly, it forces us to be entirely explicit about our interpretations. This can have tremendous advantages in terms of both conceptual and linguistic clarity. With formalized interpretations in hand it could become much easier to pinpoint the differences and overlaps between two interpretations which might be very hard to disentangle in their informal form.

3.2.2 Notions of Logical Entailment

A crucial feature of knowledge representation systems based on DL is that they contain implicit knowledge which is made explicit through logical inferences.[5, Chapter 2] The standards of correctness of these inferences is regulated by the formal definition of the DL underpinning a formal ontology. It is a question worth asking whether the patterns of inference licensed by the DL coincide with those a human researcher would draw. Abstracting from the specific issue of logical inference, misunderstandings might arise because a computational system does not represent the world and reason about it in the same way as a particular human researcher might.

There is one feature of DL-based formal ontologies where this issue becomes very tangible, namely, the *open world assumption.* DL knowledge bases are sometimes thought of as a particular kind of database. This view can lead to modeling errors. Because, although DL knowledge bases and databases bear certain similarities and analogies — schemes in databases correspond to TBoxes and actual data entries correspond to ABoxes — their semantic interpretation is markedly different. An instance of a database represents one and only one formal interpretation consisting of objects and tuples, while an ABox represents all its models. Consequently, in DL knowledge bases an absence of information is viewed agnostically — we do not know whether it holds or not. In databases, on the other hand, an absent piece of information is interpreted as false.[5, Chapter 16] The contrary assumption in DL, that absent information is not necessarily false, is called the *open world assumption.*

It might shine through that this is the stuff that major misunderstanding and futile arguments are made of. Two researcher departing from different (formal) assumptions could obtain results that diverge radically from each other, though their treatment of the subject matter might look similar on the surface. The validity of inferences, for example, could be the focal point of very fundamental difference.[22]

[22]To provide a concrete illustration: Assume we axiomatize an informal interpretation and

These are debates we must be willing to lead as there certainly exists ample room for differences in opinion when it comes to such foundational matters. However, we do gain one clear advantage through our formalization in a specific logical language. Working formally makes the difference in logical inferences explicit. By laying it in the open, together with a complete formal explanation of our reasoning principles, a discussion about such foundational issues might turn out to be more tractable.

3.2.3 NLP in Philosophy

My final point related to methodology from a philosophical perspective concerns the use of more sophisticated NLP tools. This issue is perhaps best introduced via a concrete example: Distributional models of word meaning.

Assume that a philosopher is interested in the meaning of the relationship between the concepts *woman* and *man*.[23] She might try to use distributional models as a computational tool in her quest. Distributional models are NLP tools which formally represent the meaning of words through the context in which they have been observed. Their philosophical foundation is sometimes called the *distributional hypothesis*, vaguely inspired by Ludwig Wittgenstein's philosophy. It states that two words are similar in meaning if they occur in similar contexts.[24] Typically, word meanings are represented as points in vector spaces where spatial relations in the vector space can be interpreted as semantic relations between words.[36] Proximity in the vector space, for example, is interpreted as similarity in meaning. The words' location in the vector space is based on their occurrences in bodies of data, textual, visual or otherwise. The process of embedding a word sometimes requires fairly involved algorithms.

In order to have some success in her project of comparing the concepts *man* and *woman*, the philosopher needs to critically assess whether her research question fits this rather sophisticated machinery. She must realize, for example, that whatever formal representation she receives as output from the computer is a *descriptive*, and not a *normative*, account of what the words "woman" or "man" mean in the corpus she happens to have studied.[11] Also, this is only an accurate representation of the words' meaning under the assumption that the distributional hypothesis is accepted. Furthermore, the philosopher was interested in the *concept* "woman" or "man", but what she has in hands is a formal representation of the meaning of the *words* "woman" or "man". That this is a crucial difference should be clear to any trained philosopher, but amidst all the formality and computational ruckus the point might still go underappreciated.

What is the message of all this? Philosophical investigation that relies on increasingly sophisticated computational means runs the risk of using inadequate tools to pursue its interests. This is especially true for NLP systems which seemingly deal with the same entities as philosopher (think of *meaning*) but do so in their own particular manner. This requires philosophers to constantly check whether the formal system's notion aligns with the philosophical notion in

one of the inferences drawn from it turns out to be explicitly rejected by the primary philosophical text. Now, what if the interpreter claims inference our system just drew to be invalid, and therefore her interpretation should still stand?

[23]See Herbelot, Redecker, and Müller's excellent article where this example is taken from [38].

a desirable way. We should also remind ourselves once more that these welcome systems of support provide no end-all solutions to philosophical problems, and that their outputs should be taken with a grain of salt.

3.3 Practical Perspective

Let us put these philosophical questions aside for a moment and assume a more practical perspective on the use of the method lately described. The guiding questions here are whether the method can work and how we can make it work in particular cases. Whether it can work will have to be established by using it in actual research and the present section contains more discussion of what it would mean for the method to "work". As for the second question, concerning how it can be made to work, I can impossibly give definitive answers at the present time, as nothing has been tried thus far. I shall, however, issue some thoughts on what strikes me as promising practical steps.

Notwithstanding the philosophical concerns and caveats voiced in the previous section, my method is most likely a good one if it produces good results. In this sense the ends justify the means. Yet still, what counts as a "good" result is certainly unclear and I would be foolish to simply assume that whatever comes out of an application of the method qualifies as such. Thus, I now want to assess practical points concerning three aspects of my method: Formalization, the availability and quality of data, and the necessity to test my method comprehensively.

3.3.1 Practical Aspects: Formalization in Philosophy

As a starting point I ask myself what possible good could come out of my method. In essence, there are two potential practical benefits. First, formalization itself has been widely used as a tool of analysis in philosophy, especially since the big innovations in logic have taken hold in the 20th century. Second, modern computers can do certain tasks very quickly that no human researcher can hope to do in reasonable time.

To start with practical advantages of formalization, philosophers typically communicate their insights by writing texts in natural language. Due to the high conceptual density and the sometimes idiosyncratic use of expressions, confusions arise quite easily in philosophical debate about what a piece of philosophical writing is trying to express. Because a text could mean more than one thing, an exegetical question comes up as to how the text is supposed to be read.

Since formal languages are usually designed to be very precise and free of ambiguity, translating a piece of philosophical prose into unambiguous expressions of some suitable formal language can help scholars to converge on such an interpretation. The formalization forces them to commit to one particular interpretation which can then be displayed in the open for scrutiny. It lies in the nature of this process that there are different possible formalizations of the same text. By comparing them, one can get a clearer grasp of what the difference of opinion about the correct interpretation in fact consists in. Even if no agreement can be reached, the process of formalization thus still sharpens the discussion an the way people think about the issue at hand as it forces interpreters to commit to a reading which can then be publicly debated. Hence, formalization can give us the practical benefit of an issue's clarification, irrespectively of its further processing.

As I discussed already, the language used to formalize should fit its object, just as NLP tools should fit their objects. Systematic reconstruction of unsystematic thinking is senseless, just as a vector model of word meaning might well diverge from our understanding of the concept of meaning. Let us stay with the choice of language for a moment to argue why I would choose DL for the purpose of supporting philosophical interpretation.

As already mentioned, the DL formalism is a logical language. The use of logic is beneficial in three ways.

First off, the syntax and semantics of logical languages are precisely defined. Besides the immediate benefit of clarity, both conceptually and in communication, this also increases the interoperability of knowledge systems, as systems described in a particular logical language can be made compatible with other knowledge systems for further processing. Because the language is rigorously defined, there are clear standards as to how notions must be expressed in order to be added to or merged with an existing ontology.[31]

Second, some questions are pursued far more easily in logic. Notable examples of such question are related to computational complexity (How much time do we need to answer a certain question?) and to expressivity (Can we capture a given concept in our formalism?).

Third, within the framework of a logical language we have explicit standards of correctness of inferences. Once a group of interpreters has reached sufficient agreement on how to formalize a philosophers arguments, for example, the validity of said arguments can be checked on firmly studied grounds.

This enables us to pose fundamental question concerning logical consistency in a formally precise manner. Is the philosopher's theory as presented by the interpreter consistent? Or does it perhaps contain an internal contradiction which escaped the interpreter's eye? Also related to this is the possibility of computing the logical consequences of a given axiomatization. Would some interpretation, together with its corresponding formalization, entail unwanted consequences? Would these additional theorems fortify the theory as expounded in its formalization?[24] Such questions can be made precise and answered systematically thanks to a logical formalization.

In fact, that I can use a computer to check for inferences is one of the two reasons why I choose to use logical languages from the family of DL. DL languages are designed to balance a trade-off between expressivity and computational complexity. There are effective procedures a computer can use to answer certain question about a set of DL expressions.[25] Typically, the more expressive a DL is, the harder it is to answer questions about a set of expressions.[5]

The second reason is that DL languages are extraordinarily well-studied and widely implemented in industrial applications. They are used in industrial application, for example for very large medical informatics, and perhaps most notably underly the Semantic Web.[28][5, Chapter 13][61, pp. 138–139] Given this importance, DL languages have been studied since the 1980s both in the ivory towers of academic departments as well as in more immediate, and decidedly less ivory, proximity to actual application. There is a multitude of

[24] [54] took this approach.

[25] This website lists description logics according to various computational complexity classes: http://www.cs.man.ac.uk/ ezolin/dl/

conferences, dedicated either to theoretical research in relation to DL or to more practically-minded purposes, where researchers and practitioners can exchange the growing body of practices and knowledge of DL.[26] Furthermore, there exist a number of sophisticated and optimized ready-made software packages such as ontology editors and reasoning tools.

All of this is to say that the philosophy researcher can rely on a host of communities, handbooks, online support, related work etc. to find her way with DL. There are practical and translateable applications in other domains of research and, very conveniently, no sophisticated coding is required.

3.3.2 A Question of Data

A part of the method relies on NLP techniques to relate reasoning output back to the primary text. This part presupposes that we are in possession of a digital representation of the philosophical texts in question. The corpus should be reasonably free of error and actually represent what a human would read on a page. Unfortunately, this is rarely the case. The process of turning a physical piece of writing, say, a page in a book, into a correct machine-readable representation, say, a .txt file on a computer, can be highly-error prone and very labor intensive, which is to say expensive. Furthermore, copyright regulations can sometimes make it difficult to obtain data of this type.

On the bright side, the number of digitalized books and other documents hitherto only stored physically can only increase. Also, it is possible that the technology pipeline turning physical to digital representations will be improved in the future. Presently, however, it can by no means be guaranteed that an affordable and high-quality corpus is available of any piece of philosophical writing one lays eyes upon. Thus, data preparation might consume considerable efforts before NLP can techniques can come to full fruition.

Another difficulty related to data availability is that some NLP techniques require much, much more data than typically considered for philosophical research. To train a distributional model, for example, all the writings of one particular philosopher combined are currently not enough to get any meaningful output.

3.3.3 Validation

Let us turn to the second issue outlined at the outset of this section: Does the method work? This question summarizes formalization, reasoning, the use of NLP, and the availability of data into one aggregate question. Giving it a slight methodological twist turns it into: How do I know my method delivers correct results? If I were to simply apply my method to an open issue of philosophical interpretation and get some results, how do I know that these results are adequate or correct? Since I do not have the answer to the open issue, there is no way of verifying the method's correctness.

What I need at the very least, then, is a validation case. An exegetical question to which I am confident to have a good answer. This is sometimes called

[26] For example the International Workshop on Description Logics (31st edition in 2018), or the International Conference on Formal Ontology in Information Systems (10th edition in 2018), or the International Conference on Knowledge Engineering and Ontology Development (which is part of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management in 2018).

a gold standard. Against this gold standard I can compare the performance of my method to assess whether it indeed works as I hope it to.

In the next section I will present such validation cases to check my method's workings. The gold standard will be a hand-picked set of exegetical facts which I extract from a bigger exegetical effort currently on-going in e-Ideas. In this way I hope to give some evidence that the method, as I described it abstractly, can in fact replicate some known results.

4 Case Study

This section contains an illustration of a DL-based method for philosophical interpretation. To repeat briefly, with the DL-based method I compute logical consequences of some interpretation of a philosophical text. I then treat these consequences as testable implications of a theory which I corroborate or falsify by using a mixture of computational techniques.

The illustration consists of excerpts from our current efforts within e-Ideas to formalize and computationally process parts of Bernard Bolzano's philosophy. In the medium term, our goal is to apply the DL-based method on large open issues in Bolzano scholarship, particularly, to test the relative merit of different exegetical stances. We envision formalizing different interpretations of Bolzano's philosophy in order to compute the consequences of these hypotheses and test them. As is currently becoming apparent, this is a tricky task, not least because differing interpretations tend to depart from vastly diverging mindsets, elementary conceptualizations, and vocabulary. Bridging these aspects to settle exegetical disputes on the grounds of my method will certainly take some more time. As an illustration of the difficulties involved in expressing two dissimilar interpretations, I want to offer an insight into my work with e-Ideas researchers.

This particular case concerns Bolzano's notion of *grounding* (the philosophical content is of no importance here). Our idea was to formalize each interpretation and then apply my method to these hypothesis individually. Since I am not a Bolzano expert by any stretch of the imagination, the plan was for Pauline van Wierst to break down each interpretation in somewhat natural language to put me in a position to formalize them.[27] Executing this plan, however, turned out to be tough. First of all, apart even from the fact that they are presented in different styles, the two interpretations involve vastly distinct vocabulary and it is unclear whether this translates to differing opinions about the entities involved. Whereas one speaks of grounding as concerned with "collections of truths", the other mentions "the complete ground G" and some variable "c". Then, each interpretation mentions concepts which do not occur in the respective other, such as "maximal generality", "logical deducability", "proposition", or "exact deducability". Again, it is hard to know whether this is a mere matter of attaching different names to certain concepts or whether there are substantive conceptual differences. It is clearly challenging, even for a Bolzano expert, to render two diverging scholarly accounts of the same concept — grounding and its auxiliaries — in sufficiently similar terms for formalization to make sense.[28]

Nonetheless, I believe that pursuing such efforts could prove valuable despite these difficulties. As stated before, the process of working on these formalizations already deepens researchers' understandings of both their own as well as their colleagues' viewpoints. Separating deep conceptual differences from superficial, terminological ones cannot be but beneficial. Also, I am optimistic that with additional efforts and growing experience we should be able to arrive at some form of formalization and to carry out our plans.

Because this project is on our minds, we would prefer to have some reason to believe that my method delivers correct results in easier circumstances. The

[27] The result can be seen in Appendix C.

[28] For, there is still the assumption that both interpretations are about the same concept. If the two have nothing in common in the first place deciding between them is futile anyway.

open issue we eventually want to treat, of course, cannot deliver this reason as it has no accepted solution yet. Hence, there is no way of checking whether an application of my DL-based method gives us insights that are valuable or correct. Therefore, as a necessary first step I apply the method on questions to which I know the correct answers, they serve as my gold standard. By replicating known results I validate my method.

The validation case concerns aspects of Bolzano's philosophy which are uncontroversial or even trivial to Bolzano experts. Nonetheless, these aspects form the conceptual foundation of Bolzano's philosophy. Any more complicated exegetical dispute ultimately rests on them, so while they might strike experts as trivial, these aspects are certainly not unimportant in the grand scheme of things.

4.1 Validation

My validation will consist of two parts. The first part, in Section 4.3, departs from an interpretation of a part of Bolzano's philosophy which is purposely inconsistent. I should be able to detect the interpretation's inconsistency by means of the formalization and automated reasoning steps of my method alone, so going back to the corpus is not necessary.

The second part of the validation, presented in Sections 4.4, aims to test the full method. To this end I present a set of formal expressions capturing concepts and their relations in Bolzano's conceptual universe. What I show here is a subset from our larger on-going attempt to formalize Bolzano's philosophy. This subset is chosen in such a way as to be largely free of exegetical controversy in the secondary literature. The set of formulae making up the formalization mostly pertains to Bolzano's mereology as well as his theory of science. I then divide this set in more basic statements — the *test set* — and more complex statements — the *validation set*. The validation set represents logical consequences that should follow from an interpretation as captured in the test set. I computed it by hand before implementing the test set. This allows us to check whether the automated reasoning yields the consequences it is supposed to.

I implement the test set in the ontology editor Protégé and apply its reasoning tools to the test set. This gives us a set of inferred statements about Bolzano's philosophy. In case the test set and the inferred set are consistent, I do two things to assess my output: First, I compare the set of inferred statements to the validation set. Qualitatively speaking there are four possible outcomes of this comparison. The possible observations comparing inferred set statements and validations set could be a) the inferred set is a subset of the validation set, b) the inferred and validation sets could be identical, c) the inferred and validation sets could have non-overlapping parts, or d) the inferred statements could be a superset of the validation set.

Second, I assess whether the complete set of statements can be justified as plausible interpretations of Bolzano by going back to the corpus. I try to gather this evidence by a combination of traditional expert assessment and the Ariadne tool for information retrieval.

4.2 Bernard Bolzano

Bernard Bolzano lived from 1781 to 1848 and worked mostly in the city of Prague. He was a formidable logician, mathematician — his intermediate value theorem still forms part of analysis textbooks — philosopher, philanthropist as well as a Catholic cleric and professor of the doctrine of Catholic religion at the Philosophical Faculty of the University of Prague. Bolzano was an outstandingly prolific and versatile intellectual, making substantial contributions to a wide range of fields of knowledge.[48][59]

Bolzano's work is well suited to be investigated via my method for two distinct reasons: First, he has written at length on a wide range of philosophical issues, treating topics as diverse as metaphysics, mereology, ethics, philosophy of science, religion, political philosophy, and most prominently, logic.[59][48] Second, he attempted to be very systematic in his philosophizing. This trait could perhaps be attributed to his mathematical mindset and his striving for rigorous proofs. Hence, many of these topics, although distributed across his oeuvre, can be expected to exhibit tight conceptual interconnections, instead of being merely a loose jumble of ideas. It can certainly be said that he strove for a consistent philosophical theory. Thus, it should be possible to start with a small but foundational part of his overall philosophy and observe logical ramifications beyond this selection.

4.3 Inconsistent Test Case

Assume we are given a piece of exegetical writing in which an interpreter describes Bolzano as holding the following positions:

1) All appearances are contained in the realm of the mental.

2) All subjective ideas are appearances.

3) The mental realm and the logical realm are separate.

4) My mental occurrence saying that "Fuzzy logics are the be all and end all of logic." is an example of a subjective idea which is contained in the logical realm.

Taken together, these four statements should be inconsistent. If the idea expressed in 4) is indeed a subjective idea, then by 2) it is also an appearance. By 1) this then implies that this specific idea belongs to the realm of the mental, in addition to the logical realm which was expressed in 4). But 3) expresses that nothing is contained both in the logical and the mental realm, so we have a contradiction.

My method should be able to point this out.

4.3.1 Formalization

The statements 1) to 3) correspond to a TBox with these DL expressions:

$$\text{APPEARANCE} \sqsubseteq \text{MENTAL REALM}$$
$$\text{SUBJECTIVE IDEA} \sqsubseteq \text{APPEARANCE}$$
$$\text{MENTAL REALM} \sqcap \text{LOGICAL REALM} \equiv \bot$$

The formal correlates of statement 4) belongs to an ABox:

Fuzzy logics are the be all and end all of logic. : SUBJECTIVE IDEA
Fuzzy logics are the be all and end all of logic. : MENTAL REALM

4.3.2 Implementation in Protégé and Reasoning

Figure 3 depicts the implementation as an ontology in Protégé.

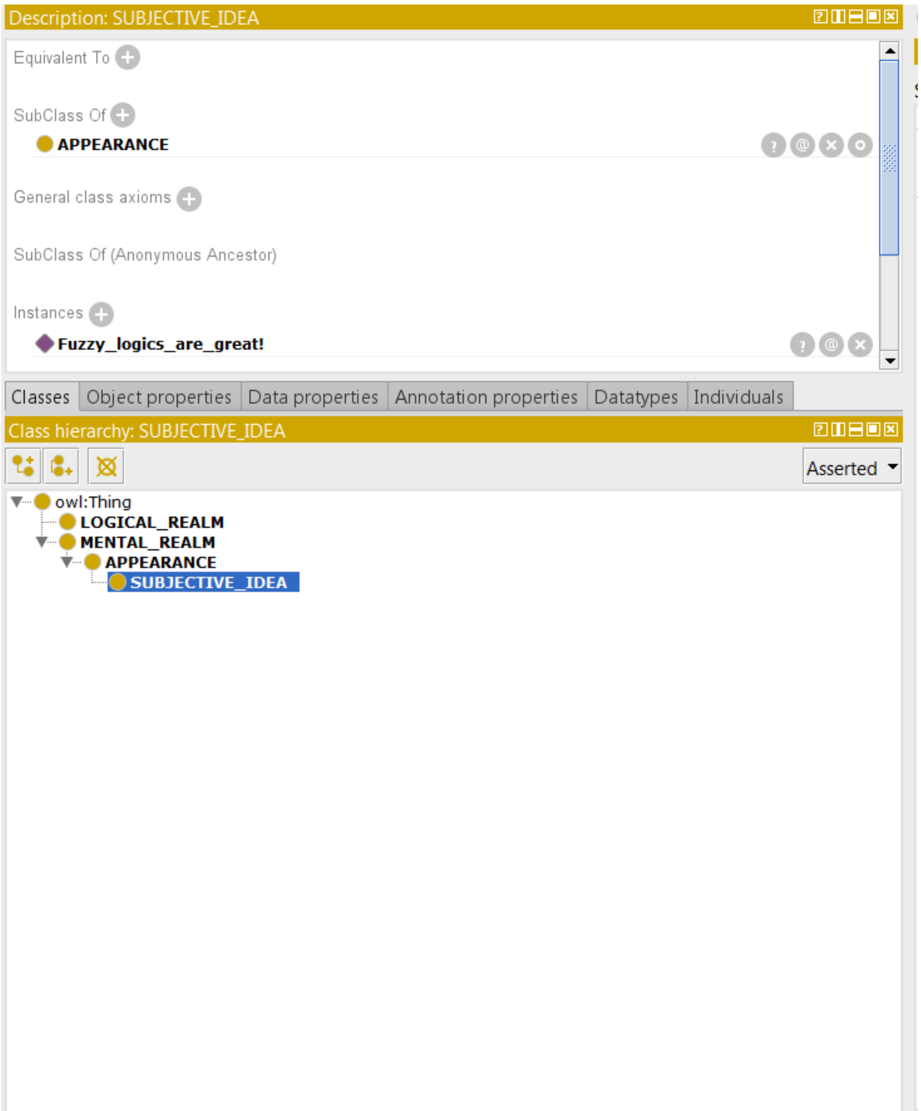

Figure 3: Ontology in Protégé

Once I apply the reasoning software Fact++, the interface flags an inconsistency, as is visible in Figure 4.

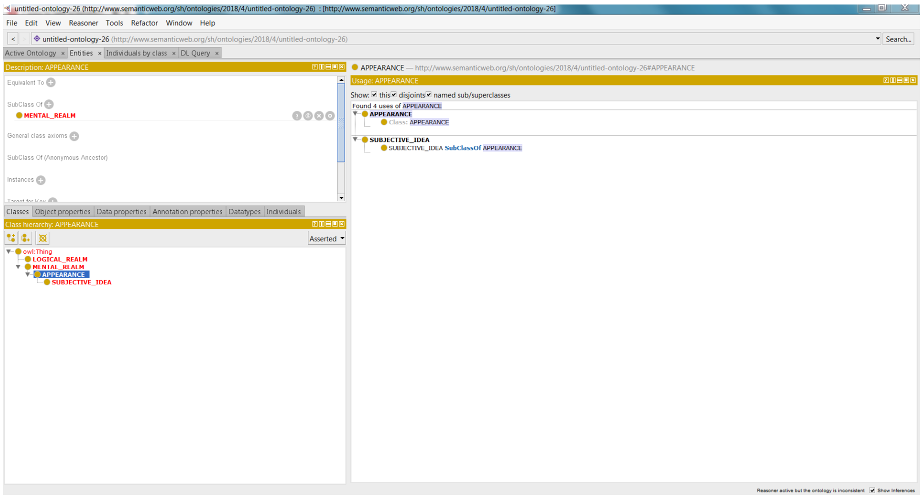

Figure 4: Inconsistency in Protégé

Protégé even opens a pop-up window stating the inconsistency and offering explanations about its origins, as is depicted in Figure 5.

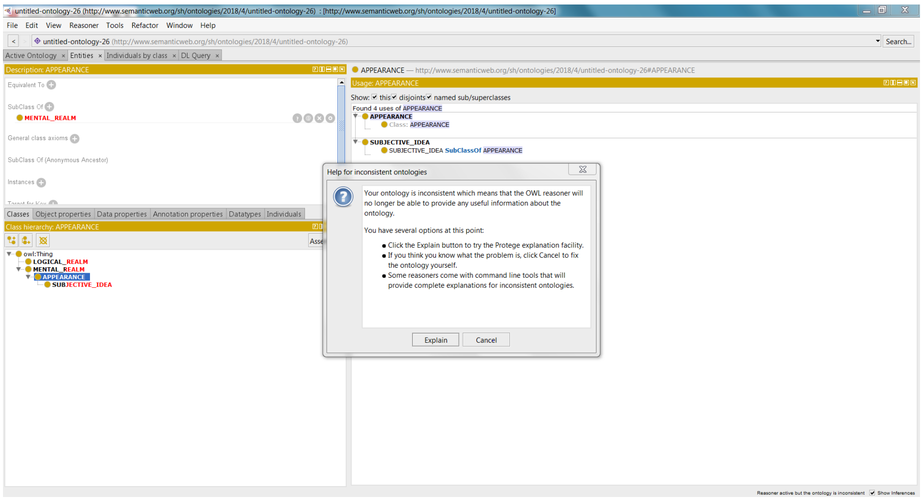

Figure 5: Warning in Protégé

4.4 Full Interpretation Test Case

In the following paragraphs I go through a full cycle of applying my method, step by step. I first present a few examples to illustrate how concepts from Bolzano's philosophy could be formalized. The full formalization which underlies this test case, as well as the validation and the inferred set to which I compare the reasoning output, can be found in the Appendix B. There you can also find a philosophical description of all the relevant concepts. Second, I implement the axiomatization and apply a reasoner to it. Crucially, the formalization captured in the test set and the validation set were completed *before* the automated reasoner was employed. I present a few examples of reasoning output and discuss how the actual reasoning output compares to the expected reasoning output in the validation set. Finally, I link the partial deductive closure back to the corpus together with a Bolzano expert.

I call the test set which contains the basic formalization $\mathcal{TEST}$, the validation set which additionally contains the inferences I expect $\mathcal{VAL}$, and the actual reasoning output which is the partial deductive closure of the test set $\mathcal{CLO}$.

4.4.1 Interpretation and Formalization

The informal interpretation of Bolzano's philosophy is taken from two sources, Arianna Betti's article *Bolzano's Universe* and Pauline van Wierst's master thesis *Salva Veritate*.[9][63] Both offer a high-level survey of Bolzano's conceptual universe. The aspects of this universe chosen for formalization in $\mathcal{TEST}$ are taken to be uncontroversial amongst Bolzano scholars.[29] To provide an impression of the process of formalization I shall now describe a handful of philosophical concepts together with their formal correlates.

Example: Formalizing the Notion of Objectual Idea In Bolzano's philosophy there is a place for genuinely logical objects such as propositions (*Satz an sich*) and ideas (*Vorstellung*). These objects, although objective and not temporally or spatially located, are said to exist as objects in some qualified way. They can be grasped in mental phenomena and expressed through language, as *subjective ideas* and *judgments* (*subjective Vorstellung* and *Urteil*). Your thought that Ghana gained independence around 60 years ago is thus a spatially and temporally located, flesh-and-blood (or rather bzzz-bzzz in brain matter) real counterpart of a logical object, namely the proposition [Ghana gained independence around 60 ago]. Likewise, your thought of Kwame Nkrumah is a real counterpart of the objective, logical object which is the idea [Kwame Nkrumah]. In Bolzano's exposition, propositions take primacy over ideas as the latter are defined only derivatively: An idea is any part of a proposition which is not itself a proposition. This is captured formally as[30]:

$$IDEA \equiv \exists is\_part\_of.PROPOSITION \sqcap \neg PROPOSITION$$

Now, ideas can stand in a relation of reference to objects (including ideas themselves). The idea [Kwame Nkrumah] refers to an object, namely, the physical person Kwame Nkrumah. In formal terms, this possibility of an idea's reference is expressed as a subsumption axiom: Any idea is such that if it stands in a relation of reference with something, then this other thing is an object of unspecified kind.[31]

$$IDEA \sqsubseteq \forall refers\_to.OBJECT$$

[29] That this is indeed so was established both in personal communication with Betti and van Wierst, and by comparison of their exegetical efforts with standard reference works such as the respective SEP articles on Bolzano and his Logic.

[30] This is expression (20) in the full formalization in Appendix B.

[31] Expression (21) in Appendix B.

Ideas can also refer to nothing. The idea [round rhombus] is such an idea. An idea that does refer is called objectual (*gegenständlich*), one that does not is called objectless (*gegenstandslos*). Formally, an objectual idea is thus characterized as an idea which stands in the relation of reference with at least one object[32].

$$OBJECTUAL\_IDEA \sqsubseteq IDEA \sqcap \exists refers\_to.OBJECT$$

4.4.2 Reasoning Output

I now assess insight into the automated reasoning step which I get from comparing the actual reasoning output, $\mathcal{CLO}$, with the set of expected inferences, $\mathcal{VAL}$, as highlighted by colors in the Appendix B. Since the validation set has been manually computed before the reasoner came into action, $\mathcal{VAL}$ can serve as an independent measuring stick for the automated reasoner's performance.

Generally speaking, the output I actually got — $\mathcal{CLO}$ — largely covered what I expected to be inferred given my test set — $\mathcal{VAL}$. This can be taken as evidence that the reasoning mechanism in the software functions to a great extent as it should. Nonetheless, there are some inferences I expected that were not in fact made, and conversely, some inferences appeared unexpectedly. From the perspective of philosophical research, a pertinent distinction could be made between interesting and non-interesting surprises. Interest originates from two distinct sources at this point. The presence or absence of an inference can be interesting either as a piece of insight into the theory I seek to axiomatize, or because it tells me something about the method I am developing.

Expectations That Were Met Many of the inferences I expected and which were in fact present in $\mathcal{CLO}$ are uninteresting in the first of these senses. They were part of $\mathcal{VAL}$ precisely because of their obviousness and hence unsurprising. An example of this are expressions such as:

$$\text{OBJECTUAL IDEA} \sqsubseteq \text{AN SICH REALM}$$
$$CONCEPT \sqsubseteq AN\_SICH\_REALM$$

These expressions merely reiterate that logical objects, such as objectual ideas and concepts, are placed in the an-sich realm.[33] In the second sense, however, their presence is a testimony to the correct workings of my method. They are what I expected, so the fact that my method (partially) gave them as output speaks for it.

Moreover, it must be kept in mind that in a real application of this method, where it is much unclearer what inferences we are supposed to expect, any inference could possibly be interesting. The fact that some of them strike us as trivial does not mean that their triviality was known to us beforehand, otherwise we would have presumably modeled them explicitly. Thus any expression color coded in black in the Appendix B is potentially informative in the first sense. Because it is as good as impossible to know beforehand what consequences are interesting, it transpires that in any real application of this method we

[32] Expression (23) in Appendix B.
[33] In fact, all expressions from (117) to (124) are of this type.

shall have to weed through the partial deductive closure, either manually or computationally[34], in order to separate the wheat from the chaff.

Unexpected Inferences By contrast, all inferences I did not expect, that is to say, all inferences that are in $\mathcal{CLO}$, but not in $\mathcal{VAL}$, are interesting in one of the two senses lately described. Some of the novel inferences are genuinely surprising as I have not thought of the connections expressed therein before. Nice examples of this are these two expressions[35]:

$$AGGREGATE \sqcap SIMPLE\_OBJECT \equiv \bot$$
$$PURE\_OBJECT \sqcap ADHERENCE \equiv \bot$$

The first of these states that no things are both aggregates and simple objects; the second states that nothing is both a pure object and an adherence. Due to their categorical nature ("There is no object such that it is A and B".) they are ideal for the next phase of my method where I look to falsify or corroborate these statements, since one counter-example in the corpus is enough to reject them.

Other inferences are, in hindsight of course, so obvious that they are unlikely to surprise any researcher. Examples here include novel inferences that I apparently simply forgot to model explicitly, such as[36]:

$$< camel^{idea}, (\text{Camel is red.})^{Proposition} >: \ is\_part\_of$$
$$< camel^{idea}, (\text{Red camel is squared.})^{Proposition} >: \ is\_part\_of$$

Both of these state that the idea [camel] is part of specific propositions, namely, the proposition [The camel is red.] and [The red camel is squared.], respectively. If nothing else, the fact that these were inferred at all highlights my carelessness in explicitly modeling role inverses.[37]

Expected But Absent Finally, there are expressions which I expected but have not shown up, such as[38]:

$$IDEA \sqsubseteq PURE\_OBJECT$$
$$PROPOSITION \sqsubseteq PURE\_OBJECT$$

The first states that all ideas are pure objects, while the second states that all propositions are pure objects. The absence of these inferences made me pause and reflect, since they should have been inferred but in fact were not. What seems to have happened in these instances is that the ontology I built works with the *open world assumption.* Recall that this means that any fact not explicitly stated is not taken to be false, but simply indeterminate. Now, a necessary condition to be a pure object is to *not* be a quality.[39] When modeling,

[34]Much as Kamps and Masuch did it in [41].
[35](Expressions 134) and (142) in Appendix B.
[36]More completely, all expressions from (153) to (160).
[37]If we model that A is composed of B, then we should also model that B is a part of A. Being a part of and being composed of something are inverse relations; flipping the order of objects implies that the other relation holds.
[38]These are expressions (128) and (129).
[39]This is modeled in (45).

however, I merely omitted to model ideas and propositions as qualities, but I did not explicitly assert that they are not qualities. But by the open world assumption, saying nothing about their being qualities or not does not warrant the inference that they are in fact not qualities.[40] What is highlighted here, then, is the difficulty of ontology modeling. This particular missing inference points towards the fact that erroneous or short-sighted modeling choices can have negative consequences and that it is quite tricky to do it thoroughly.

4.4.3 Going Back to the Corpus

In order to assess its interpretative merit, I presented $\mathcal{CLO}$ — the partial deductive closure of my axiomatization — to a Bolzano expert for assessment. This expert was Pauline van Wierst whose informal interpretation was part of the basis for this whole process. According to her understanding of Bolzano's philosophy, most of $\mathcal{CLO}$ accurately reflected his conceptual universe. However, on the top of her head she was not able to name specific passages supporting single statements in $\mathcal{CLO}$. This is understandable, given the size of Bolzano's oeuvre.

To test the interpretative theory I therefore chose to use the *Ariadne* tool.[42] My hope was to quickly find passages to support specific claims in $\mathcal{CLO}$. To do so I pursued the following strategy: First, I matched each concept employed in the formal ontology with a corresponding set of natural language labels from Bolzano's philosophy. These labels were gleaned from the informal interpretations underlying my ontology.[41] Second, I used these labels to construct multi-keyword queries for Ariadne. Together with van Wierst I then read the relevant passages — Ariadne's output — selected the pertinent, and decided whether they can be used to support or refute the formal statement in question. All of this is probably explained best by means of an example:

Say we want to test whether my formal definitions of ideas and propositions, and the relations between them, accurately captures Bolzano's writing. The relevant expressions are[42]:

$$IDEA \equiv \exists is\_part\_of.PROPOSITION \sqcap \neg PROPOSITION$$
$$PROPOSITION \sqsubseteq \exists (=3) is\_composed\_of.IDEA$$

According to this formalization, ideas are defined as those parts of propositions which themselves are not propositions. The second expression then simply spells out a necessary condition to be a proposition, namely, that of having ideas as its parts.[43] The label for the concept *idea* in Bolzano is the German "Vorstellung", *proposition* corresponds to "Satz an sich", and the *is composed of* relation to "besteht (aus)". I used these three keywords as Ariadne's input, but initially received an output, depicted in Figure 6 where passages consisting of the phrase "willkürlicher Satz" dominated.

[40]The same explanation can be offered for (143),(144),(148), and (149).

[41]Note that it is essential that the labels are collected from the very same interpreter whose conception I am formalizing, since interpreters not only disagree about the conceptual relations, but also about what natural language terms in Bolzano's writings stand for a given concept or relation.

[42]Expressions (20) and (22) in Appendix B.

[43]This is in accordance with *Bolzano's Conjecture*[18] as rendered by Betti.

Figure 6: Willkürlicher Satz

Conveniently for me, Ariadne allows users to negatively weigh a keyword when they explicitly are not interested in it. Using this facility I added "willkürlicher Satz" as a keyword, but weighed its importance negatively to filter out these irrelevant results. The query is shown in Figure 7.

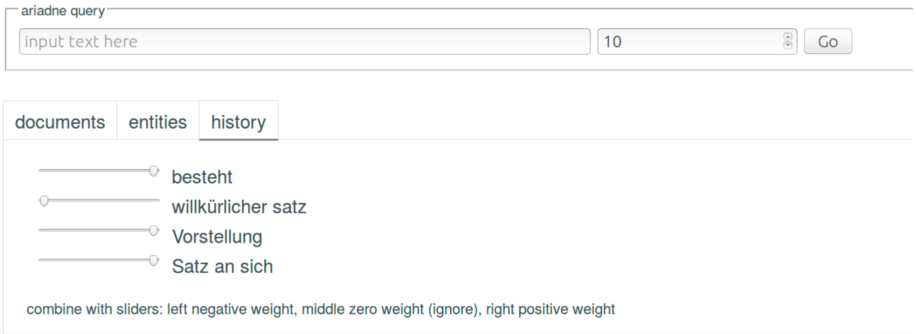

Figure 7: Keyword Weights

Searching with these parameters I found passages confirming my formalizations, as shown in Figure 8. The hit at the bottom of the screen shows a passage from a collection of Bolzano's mathematical writings that reads: "Ein jeder Satz besteht aus mehreren Theilen und diese wiefern sie nicht schon ganze Sätze sind nenne ich Vorstellungen."[44] Using my labels for the English translation this

[44]The fact that this passage is not from the *Wisenschaftslehre* poses no problem because of the previously mentioned systematicity and coherence of Bolzano's philosophy.

passage says: "Every proposition consists of multiple parts and these, if they are not whole propositions themselves, I call ideas."

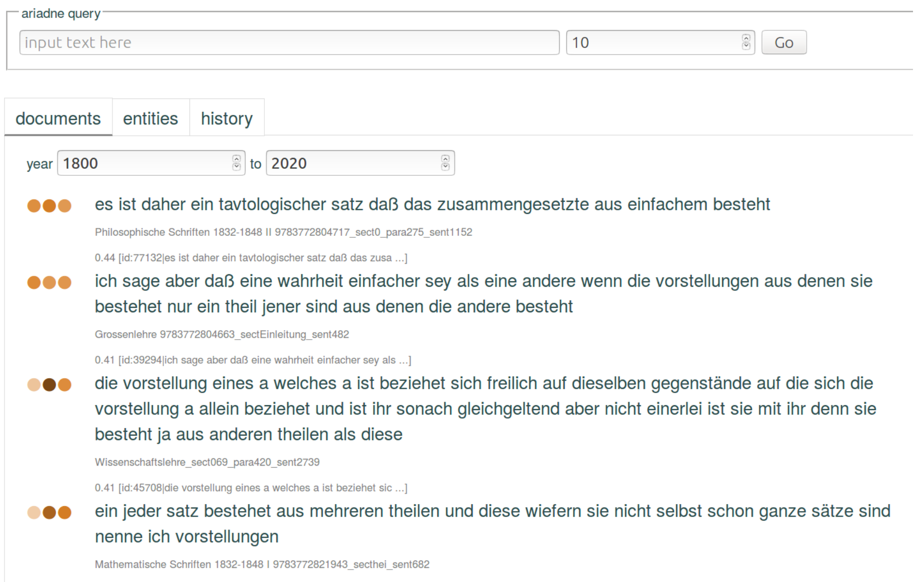

Figure 8: Confirming Evidence

I have here, thus, a confirmation of both formal expressions. The first because ideas are said to be exactly those parts of propositions which are not propositions, and the second because propositions consist of multiple parts. What is not confirmed is that these "multiple" parts are exactly three. For this, I presumably would have to mount another search. An additional piece of information I get from Ariadne is where the relevant passages can be found in Bolzano's writings. This enables one to consult the passages in their original context.

Proceeding in this way, with the Bolzano expert at my side, I was able to confirm a selection of statements in $\mathcal{CLO}$. Nonetheless, among the inferred statements there were some that called the attention of the Bolzano expert, namely[45]:

$$COMMON\_IDEA \sqsubseteq OBJECTUAL\_IDEA$$
$$SINGULAR\_IDEA \sqsubseteq OBJECTUAL\_IDEA$$
$$SYMBOLIC\_IDEA \sqsubseteq OBJECTUAL\_IDEA$$

The Bolzano expert contended that these three statements do not adequately capture Bolzano's philosophy, for she could immediately think of counterexamples in spirit of Bolzano's philosophy. She pointed out that there are examples of common, singular, and symbolic ideas with no referent.

When formalizing common, singular, objectual, and symbolic ideas, I had used the following necessary conditions to characterize them: Objectual ideas

[45]Expressions (101) to (103) in Appendix B.

are those who refer to at least one object at some point in time. Common ideas refer to at least two objects, singular ideas to exactly one. Symbolic ideas refer to other ideas. This was formally expressed thus[46]:

$$OBJECTUAL\_IDEA \sqsubseteq IDEA \sqcap \exists refers\_to.OBJECT$$
$$SINGULAR\_IDEA \sqsubseteq$$
$$IDEA \sqcap \exists (= 1) refers\_to.OBJECT$$
$$OBJECTLESS\_IDEA \sqsubseteq IDEA \sqcap \neg(\exists refers\_to.OBJECT)$$
$$COMMON\_IDEA \sqsubseteq$$
$$IDEA \sqcap \exists (\geq 2) refers\_to.OBJECT$$
$$SYMBOLIC\_IDEA \sqsubseteq \forall refers\_to.IDEA \sqcap \exists refers\_to.IDEA$$
$$SYMBOLIC\_IDEA \sqsubseteq IDEA$$

The Bolzano expert pointed out that the matter is more complicated than these formal expressions suggest. While the formalization of objectual ideas was correct, my formalization of singular and common ideas falsely suggests that they necessarily refer to some object. According to van Wierst, however, the idea [unicorn] might still be a common idea, despite its possible lack of referent. The crucial point is that *if* it were to have a referent, *then* there would be multiple of them. Analogously, the idea [the biggest unicorn] should count as a singular idea because if it were to refer to something, then only to one thing. In the expert's opinion, it was therefore wrong to model singular and common ideas as necessarily referring when they do in fact refer only contingently. In the wake of this discussion, the expert voiced doubts about the inference stating that all symbolic ideas are objectual, although she could not come up with a counterexample on the spot. I thus queried Ariadne for support. Quickly I found a passage containing a clear-cut rejection of the formal statement:

$$SYMBOLIC\_IDEA \sqsubseteq OBJECTUAL\_IDEA$$

The results of my query are depicted in Figure 9. The second hit in the list of answers states that "symbolic ideas" can be "imaginary". "Imaginary" is a label for objectlessness and hence the inferred statement that all symbolic ideas are objectual is directly refuted in this passage.

[46] These are expressions (23), (24), (25), (26), (27), and (28) in Appendix B.

Figure 9: Refuting Evidence

Subsequently, we consulted the entire paragraph from which this passage is taken in Bolzano's work. By reading this paragraph we found out that there is indeed a kind of subjective idea which is objectless. What we found out, then, is that the interpretation from which I departed and which states that there is but one kind of subjective idea was mistaken. Instead, there is a broad kind of subjective idea — coming in both the objectless and the objectual variant — and a narrow kind of subjective idea — in objectual variety only. In an updated version of my interpretation I would have to account for both of these.

Beyond the immediate corrective effect on my informal interpretation, this finding is interesting for another reason. Namely that, individually, none of the computational tools would have come up with the finding. The combination of the two computational tools was needed — automated reasoning within the formal ontology *and* Ariadne — to be able to detect shortcomings in my interpretation.

First, because the objectuality of symbolic ideas is not a fact which I modeled originally in $\mathcal{TEST}$. Rather, its explicit presence in my theory is a product of the automated reasoning procedure which I applied to my formalization. To be more precise, it results from an interaction between the definition of objectual ideas and the (erroneous) characterization of symbolic ideas. Second, once I had this piece of explicit reasoning output in hands I was able to employ Ariadne and to indicate a passage where it is directly falsified. The knowledge implicit in my formalization would have been of no further use with a tool like Ariadne.

Thus, obtaining this particular piece of insight results from a true computational team effort. Naturally, some Bolzano experts might already know about the different variants of subjective ideas and consequently this finding strikes them as trivial. Nonetheless, the fact that Pauline van Wierst did not is a testimony to the fact that Bolzano's conceptual universe is definitely too vast and complex to allow for quick and easy interpretation. In this particular case my computational method has been able to offer some support in her process towards a better understanding of Bolzano's philosophy.

5 Conclusion and Outlook

I now want to briefly recapitulate what I have done before looking beyond this thesis to consider next steps that could follow in its wake. I shall issue some general lines of future research and describe one proposal for a follow-up project.

5.1 Recapitulation

In my quest to advance the understanding of methodology as well as practical techniques of computer-supported philosophy, I have developed a method to support researchers in philosophical interpretation. This effort is firmly embedded within the larger research project e-Ideas which unites experts in philosophy and computational linguistics to combine logic-based modeling of expert knowledge about philosophical domains with NLP techniques. Specifically, I proposed ways to use DL together with automated theorem provers and NLP to support exegetical philosophy.

Equipped with a clean, machine-readable corpus of Bernard Bolzano's philosophical writings I built concrete test cases in which I applied my method. A necessary test any novel method has to pass is the replication of known results. To do so I worked with a small cluster of well-understood concepts in Bolzanian philosophy. I proceeded in the following fashion: First, I formalized fragments of interpretations of Bolzano's philosophy, as gleaned from secondary literature in Bolzanian scholarship, in a DL language. Second, I applied automated theorem provers to the formalizations, thus obtaining its partial deductive closure. Third, using NLP techniques on the Bolzano corpus, I linked the output of the reasoner back to the original text in order to obtain evidence regarding the interpretation and assessed it. Following this assessment I adjusted my original interpretation.

My ultimate goal is to apply the method in more complex scenarios, such as large fragments of controversial aspects in Bolzano's philosophy. The test case used for illustration was taken from our present attempt in e-Ideas to investigate open issues in Bolzano. This very small case has already brought one previously unnoticed aspect of Bolzano's complex philosophy to the attention of a Bolzano scholar. Eventually, we would like to be able to capture the exegetical difference between two interpretative stances in DL, so as to find evidence in support of either one of them. Much more effort will be needed to tackle these larger and more connected issues in Bolzano scholarship. Trying to do so in the future should allow us to see where the method's limitations lie, help us in developing it, and further adapt it to our particular needs. That being said, I think that the particular computational tools used in this thesis are not of true importance; rather it is about the rationale behind investigating and trying to use them. I hope to have made plausible that formal methods in conjunction with computational power can be of help to philosophical research. Thinking about them and developing them further could potentially be of great benefit.

In the remainder of this conclusion I want to take a brief look into future work that needs to and could be done to advance philosophical research along the lines proposed in this thesis. To this end, I first list a number of issues where further efforts are needed and then outline a concrete computational project for Bolzanian scholarship.

5.2 Future Work: General Remarks Regarding the Method

As pending work is concerned, there is a need for wider availability of digitalized text and other philosophically relevant data. High-quality data is of crucial importance to the success of automated processing. With low-quality input, the output has a high chance of being low quality as well. In the case of digitalized books low quality can mean, for example, faulty character recognition, footnotes that are placed in the middle of the main body of text, or end-of-line splits spilling over into the corpus. All of these are errors that accrue throughout the automated process of digitalization since a human typing the text letter for letter would presumably not make these mistakes. Human labor, however, can be very expensive and thus machines are preferable, especially considering the vast amounts of physical texts that need digitalizing.[47]

As data driven methods gain in importance in a variety of research areas, it is conceivable that improvements will be made. These improvements could take the form of technical innovation — substantially better general-purpose OCR, for example — or of institutional or legal circumstances — such as revisions of copyright laws under which certain documents are available. Keeping an eye on developments of these stages could be important for the philosopher interested in computational methods.

The same point goes for possible improvements of NLP techniques as well as the development of formal ontologies and their reasoning facilities. If the field makes advances new computational projects could come within reach for philosophy.[47]

Another area where improvements could be made are ontology's reasoning facilities. On a very fundamental level, only little is known about the limits of tractable computation.[4, pp. 84–84] Pending novel theoretical results — such as precise lower bounds — on the reasoning complexity of certain computational tasks, we can at least retain hope to take steps in this direction. If novel and better reasoning algorithms are developed, these computational methods should benefit and gain in importance. Philosophers planning to work with computational methods will have to keep an eye on technical developments here too.

Technological advances aside, my method's effectiveness would benefit from clearer and more elaborate standards of formalization. If a number of researchers, for example a group of Bolzano scholars, could agree on their conceptual stock as well as their nomenclature, then computational approaches like the one I proposed could certainly benefit. Earlier I have indicated how difficult it is to find the right set of formal terms in which to render two different philosophical interpretations. Attaining a deeper understanding of how such a shared language can be facilitated could go a long way in making my method

[47] Just one possible form this could take is illustrated by an example of a recent development in NLP which has the potential to deal with small size of philosophical corpora. When there is not enough training data, semantic vector models cannot be properly trained because the relevant words occur too few times throughout the corpus. In [37] Herbelot and Baroni pre-train a semantic vector model on other another, bigger, corpus before training it on the actual, small, corpus of interest. They let the second training phase count for much more than the first one, so that the meaning representations still shifts considerably according to the second corpus. In this way the model is fine-tuned in the second stage, on the relevant corpus. Techniques along these lines might make it possible in the future to have distributional models represent philosophical concepts.

more effective. In the last remaining section of this thesis I will therefore describe a possible project for Bolzanian scholarship. By fostering the mutual understanding of different perspectives on Bolzano's philosophy, and by trying to find a unified set of terms to express different interpretations, this project could be a concrete next step.

5.3 Future Work: A Concrete Proposal

Two types of real-life applications of formal ontologies are, one, endowing large corpora with layers of metadata and, two, building knowledge bases to model areas of human knowledge. The first of these two is used to make very large amounts of information — such as text on the internet — more manageable by structuring it conceptually, while the second corresponds to initiatives like the healthcare terminology SNOMED, for example, which looks to "represent clinically relevant information consistently, reliably and comprehensively".[39] As seen in the related work section, there have already been first attempts to build ontologies for philosophy with either of these two ideas in mind. Specific works, such as Dante Alighieri's oeuvre or Ludwig Wittgenstein's *Tractatus Logico-Philosophicus* have already been equipped with layers of metadata for scholarly or educational purposes.[6][53] Also, efforts to make the SEP more navigatable by describing its content in machine-readable form belong to this type of initiative.[49]

Following the precedent of the research just mentioned, the Bolzano corpus could benefit from a layer of metadata. For traditional philosophical standards its 11'000-page length is decidedly sizeable and sometimes a serious hindrance. Even with the somewhat sophisticated statistics-based tools used by the e-Ideas group, its size presents quite a challenge when looking for specific passages that are relevant to a given topic. Given the additional fact that the conceptual universe of Bolzano's philosophy is wide and complex, it can be hard to keep a good overview over it. Thinking back to the diverging interpretations of Bolzano's notion of grounding, getting a clearer understanding of what concepts are even relevant in Bolzano's philosophy — according to one interpreter or another — could prove helpful.

As a concrete remedy to these issues, I suggest to build an annotated ontology of Bolzano's philosophy reflecting scholarly knowledge. This could take the form of a formal ontology, i.e. a set of DL formulae, where each DL formula is annotated with the corpus passages supporting it.

Ideally the formal machinery to build this ontology would be flexible enough to account for diverging views about Bolzano's philosophy. From a synchronic viewpoint, different experts today might entertain different opinions about Bolzano's philosophy; diachronically speaking, the interpretation of Bolzano's philosophy might have varied over time and across geographical location. For some scholarly projects, certainly for researchers working on the history of ideas, it would be desirable to possess machine-readable representations of different interpretations along these dimensions. Having such explicit accounts of an interpretation's essence at a given time, in a given location, by a given interpreter, harbors tremendous potential for the study of history of philosophy.

In fact, there are efforts underway to use the model *Lemon* (Lexicon Model for Ontologies), originally designed for the semantic web, towards similar purposes.[45] Lemon is built to allow modelers to link lexical knowledge — such as spelling variants or morpho-syntactic information in a corpus — to conceptual knowledge as captured by a formal ontology. Depending on the context — location, time, writer, etc. — the same string of characters in a corpus can thus be linked to different concepts in the ontology.[48] One can exploit this feature to capture diverging interpretations of Bolzano's philosophy by making the formal interpretation of a certain word or passage in the Bolzano corpus a function of its context. Hence, one can explicitly model that, for example, occurrences of the string "Abfolge" correspond to one formalization in the context of a given 21st century philosopher working in Amsterdam, but used to correspond to a different formalization in the context of a given mathematician working at the beginning of the 20th century in Prague.

Such an ontology presents multiple advantages to expert researchers and newcomers to Bolzano's philosophy alike. It can serve as a look-up tool because each context-dependent claim about the concepts in Bolzano's philosophy and their interrelations point to the relevant corpus passages, by way of their annotations. This could not only speed up research, but be used to get a better overview over differences of interpretation. By having explicit representations of diverging formalizations, each with their own annotations pointing out its evidence, non-experts working their way into the discussion could quickly get an overview over different interpretations across time, space, and person as well as the respective support in the corpus. A second advantage lies in the potential of applying reasoning facilities to formalizations, an option thoroughly discussed and practically demonstrated already in this thesis. A third potential benefit of having a formal representation of Bolzano's philosophy is that his philosophical notions could be compared to similar notions of other philosophers in a more direct manner. Questions such as: "Does Immanuel Kant defend the same notion of *analyticity* as Bolzano?" might gain tractability. A fourth and final advantage consists in the fact that students of Bolzano's philosophy could get an overview over its conceptual structure and the most relevant passages very quickly. Such an explicit and highly structured account thus harbors great potential as a teaching facility.

Of course, there are various challenges to building such an ontology. First of all, it requires both philosophical know-how and some degree of literacy in formalisms and computer science. These requirements make it likely that team efforts should gain in importance as philosophy turns more computational. A second concern could be that some philosophical notions defy formal description. Although no general answer can be given to this last point here and it will have to be assessed case by case, I have previously pointed to attempts to deal with expressivity concerns. A final worry could be that formalizing Bolzano's philosophy yields no immediately publishable results and only becomes valuable down the line. Although this might be a legitimate concern, the existence of projects like e-Ideas testifies to the fact that some of the necessary steps are already being taken. As the number of such computationally-minded projects in philosophy grows, it should get more attractive to become part of this community in philosophical research.

[48] My inspiration for this use of Lemon is derived from [26].

References

[1] Sunitha Abburu. "A survey on ontology reasoners and comparison". In: *International Journal of Computer Applications* 57.17 (2012).

[2] Theodora Achourioti and Michiel Van Lambalgen. "A formalization of Kant's transcendental logic". In: *The Review of Symbolic Logic* 4.2 (2011), pp. 254–289.

[3] Emhimed Alatrish. "Comparison some of ontology editors". In: *Journal of Management Information Systems* 8.2 (2013), pp. 18–24.

[4] Sanjeev Arora and Boaz Barak. *Computational complexity: A modern approach*. Cambridge University Press, 2009.

[5] Franz Baader. *The description logic handbook: Theory, implementation and applications*. Cambridge University Press, 2003.

[6] Valentina Bartalesi and Carlo Meghini. "Using an ontology for representing the knowledge on literary texts: The Dante Alighieri case study". In: *Semantic Web* 8.3 (2017), pp. 385–394.

[7] Michael Baumgartner and Timm Lampert. "Adequate formalization". In: *Synthese* 164.1 (2008), pp. 93–115.

[8] Anna Bellomo et al. "Bolzano's semantic space: Philosophical investigations using an information retrieval tool". In: *Clarin 2018 (Abstract)*. Pisa, Italy, Oct. 2018. URL: https://1drv.ms/b/s!AgPq3zEkkYuOiO9KUaYXHj-UDy1idw.

[9] Arianna Betti. "Bolzano's universe". In: *Categories of Being: Essays on Metaphysics and Logic* (2012), pp. 167–190.

[10] Arianna Betti. "Ideas at scale – towards a computational history of ideas (e-Ideas)". In: *NWO - Grant Application Form (Full Proposal)* (2015).

[11] Arianna Betti. "On Haslanger's focal analysis". In: *Krisis* 1 (2014), pp. 13–18.

[12] Arianna Betti and Hein van den Berg. "Modelling the history of ideas". In: *British Journal for the History of Philosophy* 22.4 (2014), pp. 812–835.

[13] Arianna Betti et al. *History of philosophy in ones and zeroes*. 2018.

[14] Ulrich Blau. *Die dreiwertige Logik der Sprache*. Walter de Gruyter, 1978.

[15] Georg Brun. "Die richtige Formel, Philosophische Probleme der logischen Formalisierung". In: (2004).

[16] Cameron Buckner, Mathias Niepert, and Colin Allen. "From encyclopedia to ontology: Toward dynamic representation of the discipline of philosophy". In: *Synthese* 182.2 (2011), pp. 205–233.

[17] Walter A. Carnielli and Joao Marcos. "Ex contradictione non sequitur quodlibet". In: *Bulletin of Advanced Reasoning and Knowledge* 1 (2001), pp. 89–109.

[18] Ettore Casari. *Bolzano's logical system*. Oxford University Press, 2016.

[19] Gobinda G. Chowdhury. "Natural language processing". In: *Annual Review of Information Science and Technology* 37.1 (2003), pp. 51–89.

[20] David Tena Cucala, Bernardo Cuenca Grau, and Ian Horrocks. "Consequence-based reasoning for description logics with disjunction, inverse roles, number restrictions, and nominals". In: *arXiv preprint arXiv:1805.01396* (2018).

[21] Donald Davidson. "Radical interpretation". In: *Dialectica* 27.3-4 (1973), pp. 313–328.

[22] Richar L. Epstein. *The semantic foundations of logic. Vol. 2: First-order logic*. 1994.

[23] Richard L. Epstein. "The semantic foundations of logic". In: *The Semantic Foundations of Logic Volume 1: Propositional Logics*. Springer, 1990, pp. 315–321.

[24] Katrin Erk. "Vector space models of word meaning and phrase meaning: A survey". In: *Language and Linguistics Compass* 6.10 (2012), pp. 635–653.

[25] Antske Fokkens et al. "BiographyNet: Extracting relations between people and events". In: *arXiv preprint arXiv:1801.07073* (2018).

[26] Antske Fokkens et al. *On the semantics of concept drift: Towards formal definitions of semantic change*. 2016.

[27] Katrin Fundel, Robert Küffner, and Ralf Zimmer. "RelEx—Relation extraction using dependency parse trees". In: *Bioinformatics* 23.3 (2006), pp. 365–371.

[28] Pierdaniele Giaretta and Nicola Guarino. "Ontologies and knowledge bases towards a terminological clarification". In: *Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing* 25.32 (1995), pp. 307–317.

[29] Yoav Goldberg and Omer Levy. "word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method". In: *arXiv preprint arXiv:1402.3722* (2014).

[30] Pierre Grenon and Barry Smith. "Foundations of an ontology of philosophy". In: *Synthese* 182.2 (2011), pp. 185–204.

[31] Thomas R. Gruber. "Toward principles for the design of ontologies used for knowledge sharing?" In: *International Journal of Human-computer Studies* 43.5-6 (1995), pp. 907–928.

[32] Nicola Guarino and Christopher Welty. "Evaluating ontological decisions with OntoClean". In: *Communications of the ACM* 45.2 (2002), pp. 61–65.

[33] Nicola Guarino, Christopher Welty, et al. "Towards a methodology for ontology-based model engineering". In: *Proceedings of the ECOOP-2000 Workshop on Model Engineering*. 2000.

[34] Giancarlo Guizzardi and Veruska Zamborlini. "Using a trope-based foundational ontology for bridging different areas of concern in ontology-driven conceptual modeling". In: *Science of Computer Programming* 96 (2014), pp. 417–443.

[35] Zhou GuoDong et al. "Exploring various knowledge in relation extraction". In: *Proceedings of the 43rd annual meeting on association for computational linguistics*. Association for Computational Linguistics. 2005, pp. 427–434.

[36] Abhijeet Gupta et al. “Distributional vectors encode referential attributes”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 2015, pp. 12–21.

[37] Aurélie Herbelot and Marco Baroni. “High-risk learning: Acquiring new word vectors from tiny data”. In: *arXiv preprint arXiv:1707.06556* (2017).

[38] Aurélie Herbelot, Eva Von Redecker, and Johanna Müller. “Distributional techniques for philosophical enquiry”. In: *Proceedings of the 6th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. Association for Computational Linguistics. 2012, pp. 45–54.

[39] SNOMED international. *What is SNOMED CT?* 2018. URL: `https://www.snomed.org/snomed-ct/what-is-snomed-ct` (visited on 06/10/2018).

[40] Jaap Kamps. “On criteria for formal theory building: Applying logic and automated reasoning tools to the social sciences”. In: *AAAI/IAAI*. 1999, pp. 285–290.

[41] Jaap Kamps and Michael Masuch. “Partial deductive closure: Logical simulation and management science”. In: *Management Science* 43.9 (1997), pp. 1229–1245.

[42] Rob Koopman et al. “Ariadne’s thread: Interactive navigation in a world of networked information”. In: *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. ACM. 2015, pp. 1833–1838.

[43] Sandra Kübler, Ryan McDonald, and Joakim Nivre. “Dependency parsing”. In: *Synthesis Lectures on Human Language Technologies* 1.1 (2009), pp. 1–127.

[44] Hector J Levesque, Philip R Cohen, and José HT Nunes. “On acting together”. In: *AAAI*. Vol. 90. 1990, pp. 94–99.

[45] John McCrae et al. “Interchanging lexical resources on the Semantic Web”. In: *Language Resources and Evaluation* 46.4 (2012), pp. 701–719.

[46] Jean-Baptiste Michel et al. “Quantitative analysis of culture using millions of digitized books”. In: *Science* 331.6014 (2011), pp. 176–182.

[47] Shunji Mori, Ching Y. Suen, and Kazuhiko Yamamoto. “Historical review of OCR research and development”. In: *Proceedings of the IEEE* 80.7 (1992), pp. 1029–1058.

[48] Edgar Morscher. “Bernard Bolzano”. In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Fall 2014. Metaphysics Research Lab, Stanford University, 2014.

[49] Mathias Niepert, Cameron Buckner, and Colin Allen. “A dynamic ontology for a dynamic reference work”. In: *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital libraries*. ACM. 2007, pp. 288–297.

[50] Catarina Dutilh Novaes. *Formal languages in logic: A philosophical and cognitive analysis*. Cambridge University Press, 2012.

[51] Natalya F. Noy, Deborah L McGuinness, et al. *Ontology development 101: A guide to creating your first ontology*. 2001.

[52] James A. Overton. "'Explain' in scientific discourse". In: *Synthese* 190.8 (2013), pp. 1383–1405.

[53] Michele Pasin and Enrico Motta. "Ontological requirements for annotation and navigation of philosophical resources". In: *Synthese* 182.2 (2011), pp. 235–267.

[54] Gábor Péli et al. "A logical approach to formalizing organizational ecology". In: *American Sociological Review* (1994), pp. 571–593.

[55] Riccardo Pinosio. "The logic of Kant's temporal continuum". PhD thesis. Institute for Logic, Language, and Computation, 2017.

[56] Riccardo Pinosio and Michiel Van Lambalgen. "The logic and topology of Kant's temporal continuum". In: *The Review of Symbolic Logic* 11.1 (2018), pp. 160–206.

[57] Gian-Carlo Rota. "The pernicious influence of mathematics upon philosophy". In: *Synthese* 88.2 (1991), pp. 165–178.

[58] Mark Sainsbury. *Logical forms (2nd ed.)* 1991.

[59] Jan Šebestik. "Bolzano's logic". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Fall 2017. Metaphysics Research Lab, Stanford University, 2017.

[60] Martin Stokhof. "Hand or hammer? On formal and natural languages in semantics". In: *Journal of Indian Philosophy* 35.5-6 (2007), pp. 597–626.

[61] Frank Van Harmelen, Vladimir Lifschitz, and Bruce Porter. *Handbook of knowledge representation*. Vol. 1. Elsevier, 2008.

[62] Shenghui Wang and Rob Koopman. "Semantic embedding for information retrieval." In: *BIR@ ECIR*. 2017, pp. 122–132.

[63] Pauline van Wierst. "Salva veritate". PhD thesis. VU University Amsterdam, 2013.

[64] Pauline van Wierst et al. "Phil@Scale: Computational methods within philosophy". In: *Proceedings of the Third Conference on Digital Humanities in Luxembourg with a Special Focus on Reading Historical Sources in the Digital Age. CEUR Workshop Proceedings, CEUR-WS.org*. Ed. by Lars Wieneke et al. Vol. 1681. Aachen, 2016. URL: http://ceur-ws.org/Vol-1681/Wierst_et_al_Philo_at_scale.pdf.

A
A Brief Introduction to Description Logics

For this introduction to descriptions logics (DL) I rely mostly on *The Description Logic Handbook*, especially, Section 2.2.[5] This handbook makes for an excellent resource both for a broad introduction to DLs while also containing a number of in-depth chapters on specific advanced topics.

Description logic (DL) encompasses a family of logics which allows modelers to capture information about a domain of interest by building a *knowledge base* (KB).[49] The KB consists of two components: The *Terminological Box* (TBox) which introduces the vocabulary with which the domain of interest is described; and the *Assertional Box* (ABox) which makes assertions about specific, named individual objects in the domain of interest by using the vocabulary in the TBox.

The terminology provided by the TBox, in its turn, consists of two things: *Concepts* which stand for sets of individual objects; and *roles* which denote (usually) binary relationships between individual objects. In what follows, classes are denoted by capital letters and names — eg. $A, B, PROPOSITION, INTEGER$ — and individuals by smaller case letters and names — eg. a, b, p, two. The class which encompasses all individuals in a domain universe is denoted by $\top$, the empty class by $\bot$.

Roles and concepts in the TBox are either called *atomic* when they are introduced without definition, or *complex* in case they are built up from atomic ones via formal constructors. In order to facilitate the construction of complex expressions from atomic ones, DLs provide modelers with a set of syntactic operators. DL languages are equipped with model-theoretic semantics and they are differentiated by their expressive powers and limitations.

Most DLs are purposely selected decidable fragments of first-order predicate logic. Generally, the main reason to opt for DL above first-order predicate logic in knowledge modeling tasks is the decidability [50] of a number of important inference problems.[61, Chapter 3] I shall describe some of the most important inference problems in the DL context later on.

Before providing the formal definition of syntax and semantics of the DL language used in this thesis, I provide a small number of examplatory formal expressions, together with short explanations of their intended meaning, in order to illustrate to people unfamiliar with such formalisms how they can be employed.

A.1 Example Expressions

Given here are examples of expressions that could occur in a TBox whose domain of interest are numbers and their relations. These are general axioms that describe relationship between different entities — such as the prime numbers or

[49] Knowledge bases are also known as *formal ontologies*. I shall switch between the two expressions.

[50] There are known, effective algorithms to solve these problems.

the natural numbers, for example — in the ontology.

$$PRIME \sqsubset NATURAL\_NUMBER \tag{1}$$
$$EVEN \sqsubseteq NATURAL\_NUMBER \tag{2}$$
$$EVEN \sqcap ODD \equiv \bot \tag{3}$$
$$NATURAL\_NUMBER \equiv EVEN \sqcup ODD \tag{4}$$

Expression (1) states that the class of prime numbers is strictly contained in the natural numbers, that is to say, all numbers that are prime also belong to the natural numbers. (2) states that all things which belong to the class of even numbers also belong to the class of natural numbers. The possibility that they are the same is left open by (2). Expression (3) states that nothing is both an even and an odd number. To spell it out: The intersection between even and odd numbers — things that belong both to the class of even and to the class of odd numbers — is empty. Finally, (4) defines the concept of natural number. It states that the class of natural numbers is exactly equal to the class of things that are *either* even *or* odd numbers. This expression is a definition because it introduces a symbolic name $NATURAL\_NUMBER$ for a complex description consisting of two simple concepts ($EVEN$ and ODD).

The next one is not a statement about the relationship between classes, but about a specific individual, namely the number 9.

$$9 : ODD \tag{5}$$

(5) states that an individual, named 9, belongs to the class of the odds.

Concerning the roles — relationships holding between individuals — the following expressions could occur in a TBox of an ontology about classic Greek mythology. The role is_parent_of holds between the individuals $zeus$ and $herakles$ (note that the order is important). By using such roles together with a sort of existential and universal quantifiers one can describe the class of things which have at least one child, i.e. $PARENT$, or the class of things which only have humans as parents, i.e. $HUMANS$.

$$PARENT \sqsubseteq \exists is\_parent\_of.\top \tag{6}$$
$$HUMAN \sqsubseteq \forall is\_child\_of.HUMAN \tag{7}$$
$$(zeus, herakles) : is\_parent\_of \tag{8}$$
$$CHILD \sqsubseteq \exists(=2)is\_conceived\_by.PARENT \tag{9}$$

(6) states the the class of parents is subsumed by the class of things for which there exists at least one thing with which the (first) thing stands in the relation of being-a-parent-of. To put it less convolutedly, parents are a subclass of the things which are the parents of something. Expression (7) states that the class of humans is a subclass of the things for which it holds that *all* the things they are children of are humans too. (8) states that the ordered pair of individuals Zeus and Herakles stands in the is_parent_of relation, that is, Zeus is Herakles' father. Finally, (9) states that the class of children is a subset of the class of things which have *exactly* two parents. Depending on the description logic one is using, there are many such devices to restrict the quantities of role-relationships.

A.2 Formal Syntax and Semantics

In terms of syntax, description logic languages contain two basic types of expressions — *atomic concepts* and *atomic roles*. *Complex concepts* are built up inductively from these basic types by means of *concept constructors*. *Concept descriptions* comprise both atomic concepts and complex descriptions. For purposes of abstract exposition, the letters A and B denote atomic concepts, the letters C and D concept descriptions, and the letter R an atomic role.

Description logic languages are distinguished according to the concept constructors they foresee. One of the most basic ones is $\mathcal{AL}$ (attributive language). There is a naming scheme for DL languages, so that the name of a language typically gives away what constructors it allows. The language $\mathcal{ALEN}$, for example, is an extension of $\mathcal{AL}$ which also allows full existential quantification and number restrictions. Generally, $\mathcal{E}$ stands for an extension with full existential quantification, $\mathcal{N}$ for one with number restrictions, and $\mathcal{C}$ for one with negation of arbitrary concepts.

Here are the concept descriptions and constructors which appear in this thesis.

A	(atomic concept)
$\top$	(universal concept)
$\bot$	(empty concept)
$\neg A$	(atomic negation)
$C \sqcap D$	(intersection)
$C \sqcup D$	(union)
$\forall R.C$	(value restriction)
$\exists R.\top$	(limited existential quantification)
$\exists R.C$	(full existential quantification)
$(\geq n)R$	(number restriction)
$(\leq n)R$	(number restriction)
$(\neg C)$	negation of arbitrary concept)

The formal semantics of description logic languages is given by a set-theoretic model. The interpretation $\mathcal{I}$ consists of (non-empty) set $\Delta^{\mathcal{I}}$ together with an interpretation function. This function assigns every atomic concept A a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and every atomic role R a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Concept descriptions are inductively assigned the following interpretations:

$$
\begin{aligned}
&A^{\mathcal{I}} && \subseteq \Delta^{\mathcal{I}} \\
&\top^{\mathcal{I}} && = \Delta^{\mathcal{I}} \\
&\bot^{\mathcal{I}} && = \emptyset \\
&(\neg A)^{\mathcal{I}} && = \Delta^{\mathcal{I}} \setminus A^{\mathcal{I}} \\
&(C \sqcap D)^{\mathcal{I}} && = C^{\mathcal{I}} \cap D^{\mathcal{I}} \\
&(C \sqcup D)^{\mathcal{I}} && = C^{\mathcal{I}} \cup D^{\mathcal{I}} \\
&(\forall R.C)^{\mathcal{I}} && = \{a \in \Delta^{\mathcal{I}} | \forall b.(a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\} \\
&(\exists R.\top)^{\mathcal{I}} && = \{a \in \Delta^{\mathcal{I}} | \exists b.(a,b) \in R^{\mathcal{I}}\} \\
&(\exists R.C)^{\mathcal{I}} && = \{a \in \Delta^{\mathcal{I}} | \exists b.(a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\} \\
&((\geq n)R)^{\mathcal{I}} && = \{a \in \Delta^{\mathcal{I}} || \{b | (a,b) \in R^{\mathcal{I}}\}| \geq n\} \\
&((\leq n)R)^{\mathcal{I}} && = \{a \in \Delta^{\mathcal{I}} || \{b | (a,b) \in R^{\mathcal{I}}\}| \leq n\} \\
&(\neg C)^{\mathcal{I}} && = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}
\end{aligned}
$$

Two concept descriptions C and D are called *equivalent* if $C^{\mathcal{I}} = D^{\mathcal{I}}$. Concept equivalence is denoted by $C \equiv D$.

I often write $(= n)R$ as shorthand for $(\geq n)R$ and $(\leq n)R$ and naturally interpret

$$((= n)R)^{\mathcal{I}} \quad = \{a \in \Delta^{\mathcal{I}} || \{b | (a,b) \in R^{\mathcal{I}}\}| = n\} \tag{10}$$

A.3 Basic Reasoning

A computational system based on DL, a so-called *DL system* can use the information stored in TBox and ABox to automatically reason about their content.

Formal Definition of Reasoning Tasks Given a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, where $\mathcal{T}$ is a TBox and $\mathcal{A}$ is an ABox, then $\mathcal{K}$ is *consistent* if it has a model. A concept C is called *satisfiable* with respect to $\mathcal{K}$ in case there exists a model $\mathcal{I}$ of $\mathcal{K}$ with $C^{\mathcal{I}} \neq \emptyset$. An interpretation of this kind is called a model of C with respect to $\mathcal{K}$. It is said that a concept D *subsumes* the concept C with respect to $\mathcal{K}$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds for all models $\mathcal{I}$ of $\mathcal{K}$. Two concepts C and D are said to be *equivalent* with respect to $\mathcal{K}$ if they subsume each other with respect to $\mathcal{K}$. An individual a is an instance of a concept C with respect to $\mathcal{K}$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ holds for all models $\mathcal{I}$ of $\mathcal{K}$. A pair of individuals (a, b) is an instance of a role R with respect to $\mathcal{K}$ if $< a^{\mathcal{I}}, b^{\mathcal{I}} >\in R^{\mathcal{I}}$ holds for all models $\mathcal{I}$ of $\mathcal{K}$.

For the purpose of philosophical investigations two of the most interesting reasoning tasks a DL system can perform are determining whether a KB describing a domain of interest is consistent, and whether one concept description is more general than a second one. The latter is called the problem of subsumption. Using subsumption tests, we are able to deduce the hierarchy, from most to least general, in which the concepts of a TBox fall. This problem, as well as all the other ones introduced above, are reducible to KB consistency.[61, Chapter 3]

Another very useful service a reasoner can provide is computing the partial logical closure of a set of DL formulas. As a set of axioms grows larger, the harder it becomes to determine what implications might follow from it. Computing all consequences of a given set of axioms could be prohibitively time-consuming or simply error-prone. A reasoner can do the task efficiently and correctly. In some cases, the deductive closure of the axioms contained in the TBox might turn out to be unsatisfiable.

Yet another problem a DL system can help us with is to determine whether the assertions about individuals in an ABox are consistent, and whether the ABox's content logically implies that a specific individual instantiates some concept description.

B
Full Formalizations

B.1 Test Set — $\mathcal{TEST}$

Classes The following is a list of the concepts involved in formalization:

- Proposition
- Idea
  - Symbolic Idea
  - Objectual Idea
  - Singular Idea
  - Common Idea
  - Simple Idea
  - Complex Idea
- Judgments
- Subjective Idea
- Appearance
- Realms
  - Linguistic
  - Mental
  - An-Sich
  - Ontological
- Status of Existence
  - Real
  - Non-Existing
- Quality
  - Internal Quality
  - External Quality
- Object
  - Pure Object
  - Simple Object
- Substance
- Adherence
- Intuition
- Concept
- Aggregate

Role-Relations The following is a list of the role-relations involved in formalization. I chose to define inverse roles for the first two as well as the last one. On an intuitive level two roles are each other's inverses if one can switch the order (or direction) in which two objects are related to each other by a role and this yields the other role. Formally speaking, R' is the inverse of role R if and only if $\forall x, y < x, y >\in R \iff < y, x >\in R'$.

To provide a concrete example: The quality of being born on a Saturday *belongs to* Kwame Nkrumah if and only if Kwame Nkrumah *has* the quality of being born on a Saturday.

I have tried to find names for the inverse-role-pairs that are not too similar, so as to make them more easily distinguishable. This has lead to the rather clumsy *is composed of* and *is picked out by* role-names. Crucially in the context of Bolzano's philosophy, the is_part_of relation is *not* set-theoretic membership, but a special relationship holding between parts and wholes.

- A is a part of B. $\iff$ B is composed of A.
- A refers to B. $\iff$ B is picked out by A.
- A is mental counterpart to B. $\Longrightarrow$ B is an an sich counterpart to A.
- A belongs to B. $\iff$ B has A.

Metaphysics On a fundamental level, there are two dimensions on which any object existing in Bolzano's conception of the universe can be placed. In the first dimension, objects present in Bolzano's conception of the universe fall into four categories - or *realms*, as Betti calls them.[9] Any object whatsoever falls in the ontological realm. The ontological realm, in its turn, comprises at least the realm of mental objects, the realm of linguistic objects, and the realm of objects-in-themselves (*an sich*). Importantly, the last three realms are mutually disjoint. No object belongs to more than one of them.

In the second dimensions, objects are placed according to two ways-of-being - or two *boxes*, to use Betti's metaphor. One box contains objects that exist, have *Dasein*, are real or effective (presumably in the sense that they take part on relations of cause and effect). In the second box there are objects that are not characterized by any of the characteristics lately mentioned. Nonetheless, it is important that the objects in both boxes *are there* (*gibt es*) in some equal sense. Again, the two boxes are mutually disjoint. No objects is both real and not real. Table 1 displays the interplay of the two dimensions and mentions some

	Ontological	Linguistic	Mental	An Sich
Real/Existing/Being	eg. Word	eg. Word	eg. Judgments	$\emptyset$
Not-Existing	eg. Ideas	$\emptyset$	$\emptyset$	eg. Propositions

Table 1: Realms and Boxes

examples of types of objects falling under specific combination of categories. A $\emptyset$ denotes that there are no such objects. For example, there are no real objects an sich, but words are existing linguistic objects.

$$
\begin{aligned}
&LINGUISTIC\_REALM \sqcup MENTAL\_REALM \\
&\sqcup AN\_SICH\_REALM \sqsubseteq ONTOLOGICAL\_REALM && (11)\\
&LINGUISTIC\_REALM \sqcap MENTAL\_REALM \equiv \bot && (12)\\
&LINGUISTIC\_REALM \sqcap AN\_SICH\_REALM \equiv \bot && (13)\\
&MENTAL\_REALM \sqcap AN\_SICH\_REALM \equiv \bot && (14)\\
&REAL \sqcup NON\_EXISTING \equiv ONTOLOGICAL\_REALM && (15)\\
&REAL \sqcap NON\_EXISTING \equiv \bot && (16)
\end{aligned}
$$

$$
\begin{aligned}
&LINGUISTIC\_REALM \sqsubseteq REAL && (17)\\
&MENTAL\_REALM \sqsubseteq REAL && (18)\\
&AN\_SICH\_REALM \sqsubseteq NON\_EXISTING && (19)
\end{aligned}
$$

Ideas, Proposition and Reference Propositions are seen as primitives from which ideas are derived. An idea is any part of a proposition which itself is not a proposition.

Ideas can stand in a relation of reference to objects. If an idea refers to at least one object, then it is objectual. If it refers to nothing — such as the idea [square triangle] — then it is objectless. If objectual ideas - such as [cow], [Kwame Nkrumah], [truth], [being born on Saturday], or [Abdul's perception of sundown on July, 2nd in 1999] — refer to one object — like the idea [Kwame Nkrumah] or [Abdul's perception of sundown on July, 2nd in 1999] — they are singular ideas, if they refer to more than one object — such as the idea [cow] - they are called common. Symbolic ideas are ideas whose referents are ideas themselves.

As previously stated, ideas are conceived of as derivatives of propositions. The ideas [Kwame Nkrumah] and [being born on Saturday], which are not themselves propositions, are parts of the proposition [Kwame Nkruma has [being born on Saturday]. Also, according to the categories of realms and boxes, ideas and propositions are objects an sich.

$$IDEA \equiv \exists is\_part\_of.PROPOSITION \sqcap \neg PROPOSITION \tag{20}$$
$$IDEA \sqsubseteq \forall refers\_to.OBJECT \tag{21}$$
$$PROPOSITION \sqsubseteq \exists(=3)is\_composed\_of.IDEA \tag{22}$$
$$OBJECTUAL\_IDEA \sqsubseteq IDEA \sqcap \exists refers\_to.OBJECT \tag{23}$$
$$SINGULAR\_IDEA \sqsubseteq$$
$$IDEA \sqcap \exists(=1)refers\_to.OBJECT \tag{24}$$
$$OBJECTLESS\_IDEA \sqsubseteq IDEA \sqcap \neg(\exists refers\_to.OBJECT) \tag{25}$$
$$COMMON\_IDEA \sqsubseteq$$
$$IDEA \sqcap \exists(\geq 2)refers\_to.OBJECT \tag{26}$$
$$SYMBOLIC\_IDEA \sqsubseteq \forall refers\_to.IDEA \sqcap \exists refers\_to.IDEA \tag{27}$$
$$SYMBOLIC\_IDEA \sqsubseteq IDEA \tag{28}$$
$$IDEA \sqsubseteq AN\_SICH\_REALM \tag{29}$$
$$PROPOSITION \sqsubseteq AN\_SICH\_REALM \tag{30}$$

Appearances Objects an sich, such as propositions and ideas, can appear (as mental events) in the minds of cognitive agents. These appearances are distinct from their an sich counterparts. Appearances are located in time and space, they are real, and they require a unique bearer. Different thinking beings can have an appearance of the same idea or proposition at different times without there being multiple of the an sich objects, but no appearance can be shared by multiple agents.

The real mental counterparts of propositions are judgments, the real mental counterparts of ideas are subjective ideas. Both judgments and subjective ideas belong to the mental realm.

$$SUBJECTIVE\_IDEA \sqsubseteq \exists has\_an\_sich\_counterpart.IDEA \tag{31}$$
$$JUDGMENT \sqsubseteq \exists has\_an\_sich\_counterpart.PROPOSITION \tag{32}$$
$$SUBJECTIVE\_IDEA \sqsubseteq MENTAL\_REALM \tag{33}$$
$$JUDGMENT \sqsubseteq MENTAL\_REALM \tag{34}$$
$$JUDGMENT \sqcup SUBJECTIVE\_IDEA \sqsubseteq APPEARANCE \tag{35}$$

Simple and Complex Ideas, Intuitions, Concepts Ideas can be either simple or complex. The latter have at least two proper parts, whereas simple ideas do not have parts. Parts are always proper in Bolzano's mereology, thus the parthood-relation is anti-reflexive. An idea's extension is the set of objects it refers to. An objectless idea's extension is empty. The idea [Geert Wilders' natural blond mane] has an empty extension, while the idea [Belarusian gold medal winners at the winter Olympics 2018] has an extension consisting of Hanna Huskova and Darya Domracheva. If an idea is simple with respect to its parts and singular with respect to its extension, then it is called an intuition (*Anschauung*). An idea which is neither an intuition, nor has intuitions as parts is called a concept

(*Begriff*).

$$SIMPLE\_IDEA \sqsubseteq IDEA \tag{36}$$
$$SIMPLE\_IDEA \sqsubseteq \neg(\exists is\_composed\_of.OBJECT) \tag{37}$$
$$COMPLEX\_IDEA \equiv \exists(\geq 2)is\_composed\_of.IDEA \tag{38}$$
$$COMPLEX\_IDEA \sqsubseteq IDEA \tag{39}$$
$$INTUITION \equiv SIMPLE\_IDEA \sqcap SINGULAR\_IDEA \tag{40}$$
$$CONCEPT \equiv \neg INTUITION \sqcap \neg(\exists is\_composed\_of.INTUITION) \tag{41}$$

On the Interpretation of Objects There is a danger of confusing different notions when formalizing Bolzano's metaphysics in DL because of some terminological as well as conceptual overlap. Recall that the formal semantics of DL are defined in set-theoretic terms on a domain of interpretation. Formally speaking, this is a non-empty set of objects denoted $\Delta^{\mathcal{I}}$. The interpretation given to $\top$ — the class of all objects in the universe the description logic talks about — is $\Delta^{\mathcal{I}}$. Typically, the members of this class are called *individuals* in DL parlance. Conversely, the interpretation given to $\bot$ is simply the empty set.[5, Chapter 2]

Bolzano seems to entertain notions that are at least superficially similar to these set-theoretic objects, so we must make sure not to conflate any of them. Particularly, this goes for Bolzano's *objects* and his *ontological realm*. Let me briefly comment on the issue how Bolzano's notions relate to the formal interpretation we make of them when formalizing his philosophy. Of course, there is a multitude of ways to handle this relationship, with potentially far-reaching consequences, but I will restrict myself to describing how I did it so far and return to the issue later.

First, does Bolzano's notion of *ontological realm* correspond to the class of everything $\Delta^{\mathcal{I}}$? This is to say, is everything that exists in one way or another (read, *es gibt*) in the ontological realm? I assumed the answer of this to be negative. Thus, there could potentially be an element in the formal semantics x such that $x \in \Delta^{I}$, but $x \notin (ONTOLOGICAL\_REALM)^{\mathcal{I}}$. Concisely put, $(ONTOLOGICAL\_REALM)^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$. (Note that this is the real-deal: A proper set-theoretic subset relation. It is not the $\sqsubseteq$ expression pertaining to the syntax of DL.)

This provides a way of answering to the second question of whether absolutely anything that is somehow present in Bolzano's universe is properly called an object. If one would like the notion of *object* to be all-encompassing in this manner, then one can simply equate the interpretation of the class of objects with the class of everything present in the ontology. That is to say, $(OBJECT)^{\mathcal{I}} = \Delta^{\mathcal{I}}$. A consequence of this would be that the interpretations of $\top$ coincides with $OBJECT$, and the interpretation of $\bot$ with that of $\neg OBJECT$. Differently put, yet again, there would be nothing that is not an object.

However, I chose to formalize the notion of object in more restricted fashion. Two consideration led to this choice: I already chose to view the ontological realm to potentially be a strict subset of the whole domain of interpretation. And I would like to view objects as belonging to the ontological realm, so that everything called an object by Bolzano is part of the ontological realm. Thus, the interpretation of the class of objects is a subset of the interpretation of the class that is the ontological realm, $(OBJECT)^{\mathcal{I}} \subseteq (ONTOLOGICAL\_REALM)^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$.

To sum up:

$$
\begin{aligned}
&(\bot)^{\mathcal{I}} = \emptyset \\
&(\top)^{\mathcal{I}} = \Delta^{I} \\
&(ONTOLOGICAL\_REALM)^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \\
&(OBJECT)^{\mathcal{I}} \subseteq (ONTOLOGICAL\_REALM)^{\mathcal{I}}
\end{aligned}
$$

Objects and Qualities Every object either belongs (*zukommt*) to some other object or it does not. If it does belong to some other object, then the object is a quality (*Beschaffenheit*). If the object does not belong to any object, then it is a pure object. There are ideas of both types of object.

Every quality is an object, but not every object is a quality. Every object that is not a quality is a pure object. Propositions and ideas are examples of pure objects. Real pure objects are substances. Real qualities are adherences.

$$QUALITY \equiv \exists belongs\_to.OBJECT \tag{42}$$
$$PURE\_OBJECT \equiv \neg(\exists belongs\_to.OBJECT) \tag{43}$$
$$QUALITY \sqsubseteq OBJECT \tag{44}$$
$$PURE\_OBJECT \equiv OBJECT \sqcap \neg QUALITY \tag{45}$$
$$SUBSTANCE \equiv PURE\_OBJECT \sqcap REAL \tag{46}$$
$$ADHERENCE \equiv QUALITY \sqcap REAL \tag{47}$$

Aggregate, Menge, Sum, Property, Relation Similarly to ideas objects can be either simple or complex, complex objects being those which have (at least two) parts. Complex objects are most prominently called aggregates (*Inbegriffe*). Two special kinds of aggregates are manifolds (*Mengen*) and sums (*Summen*). The characteristic that sets manifolds and sums apart as special aggregates is that the order in which their parts occur is of no importance. The difference between sums and manifolds lies in what counts as their parts. For an aggregate that is a manifold only the immediate parts of the aggregate count as parts of the manifold. For sums the collection of its parts is closed under the parthood-relation, so the parts of any of its parts also count as parts of the sum.

There are two types of qualities ascribed to objects. Internal or absolute qualities, also called properties (*Eigenschaften*), are ascribed to objects insofar as they stand alone. Relations (*Verhältnisse*), external qualities, are qualities of aggregates. They are ascribed to objects in view of the relation they stand in with other objects. Wherever there is a relation there is an aggregate, and wherever there is an aggregate there is a relation.

$$AGGREGATE \equiv \exists(\geq 2) is\_composed\_of.OBJECT \quad (48)$$
$$SIMPLE\_OBECT \equiv \neg(\exists is\_composed\_of.OBJECT) \quad (49)$$
$$AGGREGATE \sqsubseteq OBJECT \quad (50)$$
$$SIMPLE\_OBJECT \sqsubseteq OBJECT \quad (51)$$
$$INTERNAL\_QUALITY \sqsubseteq QUALITY \quad (52)$$
$$EXTERNAL\_QUALITY \sqsubseteq QUALITY \quad (53)$$
$$EXTERNAL\_QUALITY \equiv \exists belongs\_to.AGGREGATE \quad (54)$$
$$AGGREGATE \equiv \exists has.EXTERNAL\_QUALITY \quad (55)$$

Individuals: Simple and Complex Ideas, Referents The following are statements positing concrete individuals in order to populate the ABox with ideas and referent objects.

$$camel : OBJECT \sqcap REAL \quad (56)$$
$$red : OBJECT \quad (57)$$
$$squared : OBJECT \quad (58)$$
$$has : OBJECT \quad (59)$$
$$camel^{idea} : SIMPLE\_IDEA \quad (60)$$
$$red^{idea} : SIMPLE\_IDEA \quad (61)$$
$$squared^{idea} : SIMPLE\_IDEA \quad (62)$$
$$has^{idea} : SIMPLE\_IDEA \quad (63)$$
$$(red\_camel) : PURE\_OBJECT \quad (64)$$
$$(red\_camel)^{idea} : COMPLEX\_IDEA \quad (65)$$
$$< (red\_camel)^{idea}, (red\_camel) >: refers\_to \quad (66)$$
$$< (red\_camel)^{idea}, red^{idea} >: is\_composed\_of \quad (67)$$
$$< (red\_camel)^{idea}, camel^{idea} >: is\_composed\_of \quad (68)$$
$$< red^{idea}, red >: refers\_to \quad (69)$$
$$< camel^{idea}, camel >: refers\_to \quad (70)$$
$$< squared^{idea}, squared >: refers\_to \quad (71)$$
$$< has^{idea}, has >: refers\_to \quad (72)$$
$$< (red\_camel), red >: is\_composed\_of \quad (73)$$
$$< (red\_camel), camel >: is\_composed\_of \quad (74)$$

Note that 73 and 74 make explicit an assumption that complex (real) object *red camel* is composed of the simpler entities *red* and *camel*.

Individuals: Propositions The following are statements positing concrete individuals in order to populate the ABox with propositions.

$$(\text{Camel is red.})^{Proposition} : PROPOSITION \tag{75}$$

$$< (\text{Camel is red.})^{Proposition}, camel^{idea} >: is\_composed\_of \tag{76}$$

$$< (\text{Camel is red.})^{Proposition}, has^{idea} >: is\_composed\_of \tag{77}$$

$$< (\text{Camel is red.})^{Proposition}, red^{idea} >: is\_composed\_of \tag{78}$$

$$< camel, red >: belongs\_to \tag{79}$$

$$(\text{Red camel is squared.})^{Proposition} : PROPOSITION \tag{80}$$

$$< (\text{Red camel is squared.})^{Proposition}, camel^{idea} >: is\_composed\_of \tag{81}$$

$$< (\text{Red camel is squared.})^{Proposition}, has^{idea} >: is\_composed\_of \tag{82}$$

$$< (\text{Red camel is squared.})^{Proposition}, (red\_camel)^{idea} >: is\_composed\_of \tag{83}$$

$$< (\text{Red camel is squared.})^{Proposition}, red^{idea} >: is\_composed\_of \tag{84}$$

$$< (\text{Red camel is squared.})^{Proposition}, camel^{idea} >: is\_composed\_of \tag{85}$$

$$< (red\_camel), squared >: has \tag{86}$$

Note that 79 makes explicit the assumption that the object *red* somehow belongs to the object *camel* (in a de re sense). The same goes for 86. Possibly this formalization is wrong and the *belongs_to* relation should hold between ideas of things, not things themselves.

B.2 Expected Inferences — $\mathcal{VAL}$

This formalization captures consequences one expects to spring from the test set. Due to the very finite computational resources in my brain this is most likely an incomplete list. However, in absence of a higher Karat gold standard this is the best available option to compare my reasoning output.

Metaphysical Status

$$OBJECTUAL\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{87}$$

$$SINGULAR\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{88}$$

$$COMMON\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{89}$$

$$SYMBOLIC\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{90}$$

$$SIMPLE\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{91}$$

$$COMPLEX\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{92}$$

$$INTUITION \sqsubseteq AN\_SICH\_REALM \tag{93}$$

$$CONCEPT \sqsubseteq AN\_SICH\_REALM \tag{94}$$

Object Every class mentioned should be subsumed by the class $OBJECT$.

Ontological Realm Every class mentioned should be subsumed by the class $ONTOLOGICAL\_REALM$.

Mereology

$$SIMPLE\_IDEA \sqsubseteq SIMPLE\_OBJECT \quad (95)$$
$$COMPLEX\_IDEA \sqsubseteq AGGREGATE \quad (96)$$
$$PROPOSITION \sqsubseteq AGGREGATE \quad (97)$$
$$IDEA \subseteq PURE\_OBJECT \quad (98)$$
$$PROPOSITION \subseteq PURE\_OBJECT \quad (99)$$

Miscellaneous

$$OBJECTUAL\_IDEA \sqcap OBJECTLESS\_IDEA \equiv \bot \quad (100)$$
$$COMMON\_IDEA \sqsubseteq OBJECTUAL\_IDEA \quad (101)$$
$$SINGULAR\_IDEA \sqsubseteq OBJECTUAL\_IDEA \quad (102)$$
$$SYMBOLIC\_IDEA \sqsubseteq OBJECTUAL\_IDEA \quad (103)$$

Disjointedness The class $IDEA$ and all its subclasses are disjoint from the class $REAL$ and from $PROPOSITION$. To give an explicit formal example for $CONCEPT$ and $INTUITION$.

$$REAL \sqcap INTUITION \equiv \bot \quad (104)$$
$$REAL \sqcap CONCEPT \equiv \bot \quad (105)$$
$$PROPOSITION \sqcap INTUITION \equiv \bot \quad (106)$$
$$PROPOSITION \sqcap CONCEPT \equiv \bot \quad (107)$$

Individuals

$$camel : PURE\_OBJECT \quad (108)$$
$$camel : SUBSTANCE \quad (109)$$
$$red : QUALITY \quad (110)$$
$$squared : QUALITY \quad (111)$$
$$has : PURE\_OBJECT \quad (112)$$
$$squared : EXTERNAL\_QUALITY \quad (113)$$
$$(\text{Red camel is squared.})^{Proposition} : AGGREGATE \quad (114)$$
$$(\text{Camel is red.})^{Proposition} : AGGREGATE \quad (115)$$
$$(116)$$

B.3 Actual Reasoning Output — $\mathcal{CLO}$

This section compares the validation set, $\mathcal{VAL}$, with the partial deductive closure of the test set, $\mathcal{CLO}$. I adopt the following color code to highlight the differences:

- black for all formal expressions contained in $\mathcal{VAL}$ and confirmed in $\mathcal{CLO}$, that is, $\mathcal{CLO} \cap \mathcal{VAL}$. These are the consequences I predicted to follow and which indeed were inferred.

- blue for all formal expressions not contained in $\mathcal{VAL}$ but contained in $\mathcal{CLO}$, that is, $\{x|\ x \in \mathcal{CLO} \wedge x \notin \mathcal{VAL}\}$. These are the novel or unforeseen consequences of my formalization.
- red for all formal expressions contained in $\mathcal{VAL}$ but not contained in $\mathcal{CLO}$, that is, $\{x|\ x \notin \mathcal{CLO} \wedge x \in \mathcal{VAL}\}$. These are consequences that should have been inferred , but ended up not being there.

Metaphysical Status

$$OBJECTUAL\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{117}$$
$$SINGULAR\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{118}$$
$$COMMON\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{119}$$
$$SYMBOLIC\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{120}$$
$$SIMPLE\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{121}$$
$$COMPLEX\_IDEA \sqsubseteq AN\_SICH\_REALM \tag{122}$$
$$INTUITION \sqsubseteq AN\_SICH\_REALM \tag{123}$$
$$CONCEPT \sqsubseteq AN\_SICH\_REALM \tag{124}$$

Object It is inferred that $OBJECT \equiv ONTOLOGICAL\_REALM \equiv owl : Thing$. $owl : Thing$ is the class of all things.

Mereology

$$SIMPLE\_IDEA \sqsubseteq SIMPLE\_OBJECT \tag{125}$$
$$COMPLEX\_IDEA \sqsubseteq AGGREGATE \tag{126}$$
$$PROPOSITION \sqsubseteq AGGREGATE \tag{127}$$

$$IDEA \sqsubseteq PURE\_OBJECT \tag{128}$$
$$PROPOSITION \sqsubseteq PURE\_OBJECT \tag{129}$$

Miscellaneous

$$OBJECTUAL\_IDEA \sqcap OBJECTLESS\_IDEA \equiv \bot \tag{130}$$
$$COMMON\_IDEA \sqsubseteq OBJECTUAL\_IDEA \tag{131}$$
$$SINGULAR\_IDEA \sqsubseteq OBJECTUAL\_IDEA \tag{132}$$
$$SYMBOLIC\_IDEA \sqsubseteq OBJECTUAL\_IDEA \tag{133}$$

Disjointedness Except for the class of $SIMPLE\_IDEA$ and $INTUITION$ all ideas are inferred to be disjoint from $REAL$ and $PROPOSITION$.

$$AGGREGATE \sqcap SIMPLE\_OBJECT \equiv \bot \quad (134)$$
$$COMPLEX\_IDEA \sqcap SIMPLE\_OBJECT \equiv \bot \quad (135)$$
$$PROPOSITION \sqcap SIMPLE\_OBJECT \equiv \bot \quad (136)$$
$$COMMON\_IDEA \sqcap SIMPLE\_IDEA \equiv \bot \quad (137)$$
$$COMMON\_IDEA \sqcap OBJECTLESS\_IDEA \equiv \bot \quad (138)$$
$$SYMBOLIC\_IDEA \sqcap OBJECTLESS\_IDEA \equiv \bot \quad (139)$$
$$SUBSTANCE \sqcap QUALITY \equiv \bot \quad (140)$$
$$SUBSTANCE \sqcap ADHERENCE \equiv \bot \quad (141)$$
$$PURE\_OBJECT \sqcap ADHERENCE \equiv \bot \quad (142)$$

Individuals It is unclear to me why the last two items have not been correctly inferred. The other lines in red seem to stem from the open-world assumption.

$$camel : PURE\_OBJECT \quad (143)$$
$$camel : SUBSTANCE \quad (144)$$

$$red : QUALITY \quad (145)$$
$$squared : QUALITY \quad (146)$$
$$\quad (147)$$

$$has : PURE\_OBJECT \quad (148)$$
$$squared : EXTERNAL\_QUALITY \quad (149)$$
$$(\text{Red camel is squared.})^{Proposition} : AGGREGATE \quad (150)$$
$$(\text{Camel is red.})^{Proposition} : AGGREGATE \quad (151)$$

$$(red\_camel)^{Idea} : OBJECTUAL\_IDEA \quad (152)$$
$$< camel^{idea}, (\text{Camel is red.})^{Proposition} >: is\_part\_of \quad (153)$$
$$< has^{idea}, (\text{Camel is red.})^{Proposition} >: is\_part\_of \quad (154)$$
$$< red^{idea}, (\text{Camel is red.})^{Proposition} >: is\_part\_of \quad (155)$$
$$< camel^{idea}, (\text{Red camel is squared.})^{Proposition} >: is\_part\_of \quad (156)$$
$$< has^{idea}, (\text{Red camel is squared.})^{Proposition} >: is\_part\_of \quad (157)$$
$$< (red\_camel)^{idea}, (\text{Red camel is squared.})^{Proposition} >: is\_part\_of \quad (158)$$
$$< red^{idea}, (\text{Red camel is squared.})^{Proposition} >: is\_part\_of \quad (159)$$
$$< camel^{idea}, (\text{Red camel is squared.})^{Proposition} >: is\_part\_of \quad (160)$$

C
Bolzano's Notion of Grounding

These are the informal interpretations I received in order to formalize them. Both Roski and Rumberg's as well as van Wierst and Betti's interpretations concern Bolzano's notion of *grounding*.

Interpretation Stefan Roski & Antje Rumberg A collection of truths Φ is the complete ground of a truth Ψ iff Ψ is logically deducible from Φ and
(a) no truth in Φ is of a higher degree of complexity than Ψ;
(b) there is no smaller collection of truths than Phi from which Ψ is logically deducible in a way that satisfies (a) and which does not contain Ψ;
(c) for neither any truth in Φ nor Ψ there is a logically equivalent one of a lower degree of complexity.

Interpretation Pauline van Wierst & Arianna Betti G is the complete ground of c iff (two cases):

1. G consists of 1 proposition:
    - c is logically equivalent to G
    - G is less complex than c;
    - G is the simplest among all truths logically equivalent to it.

    - G consists of > 1 propositions:
    - c is the simplest among all truths logically equivalent to it;
    - each of the propositions in G is maximally general (synthetic);
    - c can be exactly deduced from G.