Operationalizing Scholarly Observations in OWL

Emilio M. Sanfilippo[1,2,*], Claudio Masolo[1], Alessandro Mosca[1] and Gaia Tomazzoli[3]

[1]*CNR ISTC Laboratory for Applied Ontology, Trento & Catania, Italy*
[2]*CESR University of Tours, France*
[3]*DSEAI, Sapienza Università di Roma, Italy*

Abstract
There is increasing attention in the Digital Humanities for information systems to document and analyze scholarly debates, a case for all being that of criticism in literature or performing arts. To make these systems available, there is a call for formal models to represent alternative perspectives about the same entity (text, music composition, artwork, etc.), even when they do not perfectly align or are explicitly incompatible. Building on previous research, we propose the core module for an ontology to model scholarly perspectives as *observations*. To enable tasks such as data classification, sharing, and reasoning, the ontology takes the advantages of formal languages such as OWL and SWRL. The ontology is developed with a modular architecture, where the presented module is the most general one for reuse across different domains. We also present a case study in literary studies to exemplify the ontology and show its potential usage.

Keywords
ontologies for the humanities, observations, scholarly interpretation, literary criticism, literature

1. Introduction

There is increasing attention in the Digital Humanities (DH) for the design of tools addressing both the *hypothetical* and *partial* nature of data in the humanities [1, 2, 3]. These two issues arise from factors such as the presence of multiple documents presenting alternative and sometimes incompatible information about the same entities, the lack or only partial availability of sources, or the presence of diverse viewpoints in the scholarly debate, to name just a few cases. To make some simple examples in historical research, scholars may find documents reporting *conflicting* data for the birth date of a person, may *not* have this information at all, or may be in situations where they are unsure about the *reliability* of the available documents. On the other hand, it is also common for scholars to *disagree* on a shared subject of study, one case for all being that of literary or art criticism [4].

To make sense of *observational data* produced through research investigation, as well as design information systems for their documentation and analysis, we need first of all models suited for their representation. Ideally, once a model in this direction is developed, different mechanisms can be used to compare alternative data about the same entities, thus representing debates with competing and conflicting perspectives. In addition, an approach along these lines could be functional in making artificial agents capable of handling plural and contrasting viewpoints.

A model for the representation of observational data in the DH must be able to face several challenges. First, considering the broad spectrum of research in the community, the model has to be designed at different levels of generality to be reusable across various application contexts. Second, scholars in the humanities, even within the same research field, can easily adopt different conceptual systems, where the intended meaning of the terms they use often remains implicit or only vaguely defined. Thus, a model to represent and reason over observational data must be able to deal with such a plurality of frameworks and concepts, "operationalizing" them for computational processing [5]. Third, the model

SWODCH'24: International Workshop on Semantic Web and Ontology Design for Cultural Heritage, October 30–31, 2024, Tours, France
*Corresponding author
emilio.sanfilippo@cnr.it (E. M. Sanfilippo); claudio.masolo@loa.istc.cnr.it (C. Masolo); alessandro.mosca@cnr.it (A. Mosca); gaia.tomazzoli@uniroma1.it (G. Tomazzoli)

must be able to handle conflicting observational data while preserving logical consistency.[1]

Building on previous research [7, 8], we present an OWL ontology, including rules in SWRL, to represent and logically reason over observational data. Our purpose is to provide a formal modeling framework in the shape of a computational ontology to structure observations, making them available for further use, e.g., through techniques used in Computational Literary Studies (CLS) [9].

The paper is structured as follows. In Section 2 we present some insights on how we conceive observations. The core module of our ontology is discussed in Section 3, where we also present some methodological principles for its use. Section 4 exemplifies how the ontology can be applied to literary studies, these being currently the main application context for our research. Section 5 compares the ontology with similar efforts in the DH. Finally, Section 6 concludes the paper.

2. Insights on Observations

In this section, we discuss some foundational features related to the notion of observation, borrowed from previous work [7], to lay the ground for the understanding of our ontology.

First, observations are statements about how things are categorized on the basis of the sorts of arguments one finds in literary criticism, or technical procedures like empirical measurements with Carbon-14, just to make some examples. Observations therefore have a representational nature, since they make explicit how things are described: they are *pieces of information* representing how domain entities are understood, classified, and attributed with certain properties. From this perspective, the modeling of observations allows us to take an epistemological stance on property exemplification, since representing that an entity satisfies a property means that it has been attributed with that property. It follows that observations are not necessarily *veridical*, that is, they may not correspond to reality. For instance, one may express the observation according to which a person was born on a certain date, but the observation could be incorrect, perhaps due to reliance on unreliable historical documents. In the context of this paper, we present a framework to primarily document observations without establishing criteria for their evaluation. This is because such criteria are domain-dependent and often controversial.

Second, observations can be inter-subjectively accessed, i.e. they are not scholars' introspective beliefs. This allows us to model observations as pieces of information found in texts, without the need to refer to scholars' mental states. Also, in our framework, it is possible that different scholars express the *same* observation, even by relying on different sources. Accordingly, our proposal keeps track of the provenance of observations while abstracting from both observers and observing acts.

Third, we assume that, within specific contexts, scholars share a common vocabulary for observations, namely, they share an *observational language*. As we will see, our approach employs a finite set of (basic) observation classes that are organized at different levels of generality. Each class groups observations corresponding to the attribution of a property (relation) to one (or several) individuals, e.g., an observation class collecting the attributions of the relation *being born* between persons and dates.[2]

Fourth, observations about the same entities can be compatible (at different abstraction levels), but they can also be incompatible. For instance, a scholar may observe that a person was born in 1920, whereas another may claim that the same person was born in April 1920. Assuming background knowledge about dating and calendar, these two observations are compatible, and what changes is just their timescale. However, a third scholar may claim that that person was born in February 1921, which is incompatible with the previous observations. As said, the modeling of observations has to allow documenting controversies and incompatibilities among the perspectives of domain experts.

Finally, to better understand our notion of observation, it is worth spending some words on the philosophical debate on correlated notions. Philosophers sometimes speak of *facts* as entities that are part of reality and do not depend on the epistemic states of observers [11]. For instance, if we

[1] We can think of the variety and incompatibility of conceptual systems as two features of observations modeling specific to the humanities, more than to other fields [6].

[2] The explicit modeling of observations as domain entities, and the logical axiomatization of observational languages are important differences with *annotation vocabularies*; see, e.g., [10].

consider the *proposition* 'Juliette was born on July 18, 1987', this is true because it depicts how reality is, namely, it depicts the *fact* that Juliette was born on that specific date. Some philosophers claim that facts are what makes propositions either true or false, i.e., propositions are *truth-bearers* whereas facts are *truth-makers*. In our framework, observations are neither facts nor propositions. They are not facts because they do not stand for how things are in reality. They are closer to propositions in that they are truth-bearers, i.e., the objects of statement making acts. However, some philosophers claim that propositions are abstract entities that exist even if they are not entertained by any agent [12]. Differently, observations exist only if they have been entertained; therefore, they are not *abstracta* because they are grounded on analytic, empirical, or cognitive practices (see [7] for more details).

3. Ontology Design

In this section we present the structure of our ontology. In particular, we present the core module and some methodological principles to extend it (Section 3.1), and show how it can be used to analyze observations (Section 3.2). An exemplification of the ontology to literary studies is presented in Section 4. For the sake of presentation we will use the syntax of description logic; the OWL files are available in a dedicated open-access repository.[3]

3.1. Core Module

The purpose of this module is to lay down the core modeling elements of the ontology. The core module consists of two main (disjoint) classes, i.e., Text and Observation, including a series of relations used for their formal characterization. We will introduce here three main (disjoint) subclasses of Observation: (i) *basic observation* (BasicObs), (ii) *argumentation observation* (ArgumentObs), and (iii) *source observation* (SourceObs). The last two subclasses are fundamental to represent debates and possible disagreements about the attribution of some properties (relations) to entities in the domain. To represent the entities to which an observation applies, we use the general relation hasArgument (an OWL object property), which can be further specialized, as we will see.

Texts. Texts are here intended as abstract sequences of words (characters) in one or more languages. Though we mainly refer to texts written in natural language, the proposal also applies to oral utterances or texts written in formal languages (e.g., databases, logical theories, knowledge bases). We also assume that texts differing even for a single word are not identical, although they can be similarly interpreted and can stand in various (e.g., philological or formal) relationships with each other. A text is *abstract* in the sense that it is not, e.g., the individual pattern of ink on a specific paper that would be destroyed if the paper were destroyed. The same text can have multiple "realizations"; e.g., the physical copies of Orwell's *Animal Farm* that one finds in libraries and bookshops. For the sake of generality, we do not commit to the production context of texts, namely, whether each text is produced by someone at a time. This information can be added to further extensions of our ontology, if needed.

As we will see, texts are fundamental to represent the *sources* of observations, since each observation is always linked to a text reporting the information encoded by the observation itself.

Atomic and Compound Observations. Before discussing the three main subclasses of Observation, we introduce a general *parthood* relation, defined on the whole domain, represented by the (transitive) object property partOf and its (transitive) specialization properPartOf (standing for *proper part*).[4] Such relations allow us to distinguish *compound* observations, i.e., observations composed by other observations, from *atomic* ones, i.e., observations that are not mereologically decomposable. The idea is that compound observations group several pieces of information into a single observation. More

[3] GitHub repository: https://github.com/appliedontolab/MITE.

[4] The mereology we would like to adopt is the classical atomic extensional mereology [13], which cannot be characterized in OWL.

precisely, a compound observation is intended to stand for the conjunction of all the observations that are part of it. As we will see, the distinction between atomic and compound observations is particularly relevant in the case of argumentation observations.

We introduce the subclass of atomic observations (AtomicObs) collecting all the observations having no proper parts, see (a1).[5] To simplify our framework, basic, argumentation, and source observations are all subclasses of AtomicObs (the option where such subclasses of Observation are closed under mereological sum, e.g., basic observations exclusively composed by basic observations are still basic, is technically more complex). Furthermore, we assume that all the parts of observations are observations themselves, and that observations can be part only of other observations (see (a2) and (a3)).

a1 $\mathsf{AtomicObs} \equiv \mathsf{Observation} \sqcap \forall \mathsf{properPartOf}^{-}.\bot$

a2 $\mathsf{Observation} \sqsubseteq \forall \mathsf{partOf}^{-}.\mathsf{Observation}$

a3 $\mathsf{Observation} \sqsubseteq \forall \mathsf{partOf}.\mathsf{Observation}$

Basic Observations. Basic observations are the most simple observations representing claims about domain entities, e.g., the attribution of authorship to an artwork. Basic observations do not have other observations in their arguments (in the sense of hasArgument); this means that, differently from argument and source observations (see below), they are not observations about observations. Subclasses of BasicObs can be organized into taxonomies at different levels of generalization (specificity or granularity). The core module does not include subclasses of BasicObs, since they are always domain-dependent. However, we provide here some methodological principles to specialize BasicObs, and Section 4 discusses an example relative to literary studies. Importantly, it is possible that different groups of domain experts come up with alternative subclasses of BasicObs (alternative observation languages). It could be therefore necessary to analyze and possibly align them in scenarios where data exchange and integration are relevant. In principle, one can think in terms of a *library* of multiple (partially aligned) observation languages, representing different scholarly perspectives on domain entities.

From a methodological standpoint, we assume that users begin with introducing the main subclasses of BasicObs with a fixed number of arguments, where these subclasses collect observations with the same number of arguments. Accordingly, for each main subclass of BasicObs, users need to introduce (i) an axiom establishing the number n of arguments; (ii) n specializations of hasArgument that uniquely identify its arguments, and (iii) n axioms that guarantee the existence of these arguments. Then, as said, all the specializations of a main subclass of BasicObs maintain the same number of arguments. For instance, in the case of AuthorshipObs, formulas (e1)–(e3) introduce *cardinality restrictions*, i.e., (e1) assures that instances of AuthorshipObs have *exactly* two arguments, (e2) assures that one of these arguments is an author, and (e3) that the other argument is an artwork.

e1 $\mathsf{AuthorshipObs} \sqsubseteq \exists^{=2}\, \mathsf{hasArgument}$

e2 $\mathsf{AuthorshipObs} \sqsubseteq \exists^{=1}\, \mathsf{hasArgument.Author}$

e3 $\mathsf{AuthorshipObs} \sqsubseteq \exists^{=1}\, \mathsf{hasArgument.Artwork}$

Users can specialize a subclass by imposing additional constraints on the ranges of the arguments. For instance, one could add two subclasses of AuthorshipObs, namely StatueAuthorshipObs and PaintingAuthorshipObs, grounded on the restriction of one argument to Statue and Painting (which are two subclasses of Artwork), respectively. A different kind of specialization is conceptually more interesting in the context of observation languages. Consider scholarly debates about artworks' colors or styles. One may wish to have in the ontology both the main subclass of BasicObs for colors, i.e., ColorObs, and its specializations, e.g., RedObs, BlueObs, etc. These can in turn be further specialized through, say, ScarletObs, CrimsonObs, and MagentaObs for RedObs (without having further subclasses), and similarly for BlueObs. ColorObs and its subclasses have only one argument and, quite plausibly, the range of such argument does not vary across the subclasses.[6] Note that the way in which CrimsonObs specializes

[5] Formulas marked with a_n are the axioms of the ontology, whereas r_n are its SWRL rules. All formulas with e_n are not part of the ontology and are used for examples only.

[6] It may be the case that certain colors apply only to specific kinds of entities. We do not consider this situation here.

RedObs is different from the way CarRedObs specializes RedObs: the observation CarRedObs restricts the range of RedObs to Car, whereas CrimsonObs stands for observations at a finer level of granularity, considering that the property of *being crimson* is more specific than that of *being red*. One can think, in fact, that the taxonomy introduced above for ColorObs sets the color-values and their granularity: a red-observation is more general and less informative than a scarlet-, crimson-, or magenta-observation. This can be relevant in application contexts where agents have devices with different resolutions. For instance, some agents could distinguish only general colors like red, blue, etc., while other agents may be able to distinguish more specific shades of red, blue, etc. In this case one needs to represent the granularity of observations.

These considerations have a technical impact. While for specializations like CarRedObs (similarly for StatueAuthorshipObs and PaintingAuthorshipObs) one can impose identity conditions by means of a SWRL rule like (e4), the same does not apply to RedObs. Suppose to consider (e5). In this case we cannot distinguish, for instance, red-observations obtained through devices that only distinguish the principal colors from red-observations obtained through devices that are able to distinguish different shades of red. To distinguish the two cases, one can (i) allow instances of RedObs that are instances of no subclass, i.e., RedObs is not partitioned by its subclasses; (ii) introduce identity conditions like (e5) only for the leaves of the taxonomy and for the set-theoretically difference between the set of instances of RedObs and the union of the instances of its subclasses.

e4 $\mathsf{CarRedObs}(?x), \mathsf{CarRedObs}(?y), \mathsf{hasCar}(?x, ?c), \mathsf{hasCar}(?y, ?c) \rightarrow \mathsf{SameAs}(?x, ?y)$

e5 $\mathsf{RedObs}(?x), \mathsf{RedObs}(?y), \mathsf{hasArgument}(?x, ?c), \mathsf{hasArgument}(?y, ?c) \rightarrow \mathsf{SameAs}(?x, ?y)$

Argumentation Observations. Relying on research work in argumentation theory, in particular on the bipolar argumentation framework presented in [14], argumentation observations are introduced to (partially) document interactions between observations. In particular, *support observations* (SupportObs) model positive interactions, whereas *defeat* (*attack*) *observations* (DefeatObs) model negative interactions. Argumentation observations correspond to relations between pieces of information capturing the intuition that the supporting (defeating) observation is intended to increase (decrease) the plausibility of the supported (defeated) observation. To make an example that we will better see in the case study, one may represent that a basic observation of resemblance between two literary characters is supported by observations ascribing common attributes to both characters.

Compound observations are important when it is a whole conjunction of observations that supports an observation and not each single conjunct by its own, i.e., without any part of the supporting compound observation being sufficient to support that observation. The same goes for defeat observations. Vice versa, the support (defeat) of a compound observation can always be represented (for countable scenarios) by the logical conjunction of the supports (defeats) of all its atomic parts.

Axiom (a4) characterizes the arguments of SupportObs (the axiomatization technique is the same as the one for the subclasses of BasicObs).[7] Accordingly, each instance of SupportObs is related via hasArgument to exactly two entities; one of these arguments is the supported observation, the other one is the supporting observation. Relations hasSupportedObs and hasSupportingObs are restrictions of hasArgument which apply only to observations and are intended to explicitly distinguish the supporting observation from the supported one (following the previous discussion, the range of hasSupportedObs is AtomicObs). Defeat observations have a similar axiomatization to distinguish between defeating (via hasDefeatingObs) and defeated (via hasDefeatedObs) observations.

a4 $\mathsf{SupportObs} \sqsubseteq \exists^{=2}\mathsf{hasArgument},$
$\mathsf{SupportObs} \sqsubseteq \exists^{=1}\mathsf{hasSupportedObs}, \mathsf{SupportObs} \sqsubseteq \exists^{=1}\mathsf{hasSupportingObs}$

As said in Section 2, all observations are pieces of information representing how certain entities have properties or are related. In particular, argumentation observations represent when an observation supports or defeats another observation. In this view, it is not possible to have different support

[7]For the sake of brevity, we introduce only the core aspects concerning the axiomatization of the introduced classes and object properties; readers can refer to the OWL file for a complete overview of the axioms.

observations with the same supporting and supported observations. We impose this constraint via the SWRL rule (r1).[8] Similarly for defeat observations.

r1 SupportObs$(?x)$, hasSupportedObs$(?x, ?u)$, hasSupportingObs$(?x, ?v)$,
SupportObs$(?y)$, hasSupportedObs$(?y, ?u)$, hasSupportingObs$(?y, ?v) \rightarrow$ SameAs$(?x, ?y)$

Source Observations. Source observations play a fundamental role, because they allow us to model some aspects pertaining to the *provenance* of observations, in particular, what is the source asserting (AssertObs) or rejecting (RejectObs) an observation. Following the idea that sources must be accessible in an intersubjective manner, we assume that the source of an observation is always a text. This is common in the humanities, where scholars refer to texts to point out that someone has expressed an observation. In our approach, texts also include databases and knowledge bases as sources for observations, which is more common in science. Furthermore, it is important to stress that assertions and rejections are observations themselves. This allows us to have further argumentation or source observations (making explicit the provenance of the source observations themselves) about them. In particular, source observations can be stacked up, i.e., one can recursively have texts asserting or rejecting observations about what other texts assert or reject. We will come back to this point. Here, we just want to note that our observational framework allows for the partial documentation of the subjective interpretation of a text by making explicit the observations that each interpreter associates to a text. It is possible to explicitly represent several characteristics of the contents of texts and to study the different points of view of interpreters. In this sense, source and argumentation observations, which account for the evidence the interpreters explicitly provide to argue in favor or against a given claim, allow us to partially capture the debates among scholars.

Source observations hold between two arguments, i.e., a textual source and the asserted or rejected observation, see (a5) (analougously for RejectObs). The relations hasSrc (standing for *has source*) and hasObs (*has observation*) specialize hasArgument to uniquely individuate the two arguments of source observations. Note that the assertion of a compound observation can be represented by the assertions of its atomic parts, and the rejection of a compound observation by the disjunction of the rejections of its atomic parts. Thus, we assume that hasObs has range AtomicObs.

a5 AssertObs $\sqsubseteq \exists^{=2}$hasArgument, AssertObs $\sqsubseteq \exists^{=1}$hasSrc, AssertObs $\sqsubseteq \exists^{=1}$hasObs

Similarly to the case of argumentation observations, source observations with the same source and observation arguments are identical: (r2) is the SWRL rule for AssertObs, analogously for RejectObs.

r2 AssertObs$(?x)$, hasObs$(?x, ?u)$, hasSrc$(?x, ?v)$,
AssertObs$(?y)$, hasObs$(?y, ?u)$, hasSrc$(?y, ?v) \rightarrow$ SameAs$(?x, ?y)$

To make a simple example, let us assume to have an instance o_1 of the basic authorship observation AutorshipObs. Formula (e6) represents that there is an assert observation o_2 having observation o_1 and source txt. In simpler terms, (e6) says that txt asserts o_1.[9] One could add another assert observation o_3 saying that txt' asserts o_2, as well as a reject observation o_4 saying that txt'' rejects o_3, and so on. One can therefore have chains of source observations starting from basic or argumentation observations or, more generally, from any kind of observation.

e6 AuthorshipObs(o_1), AssertObs(o_2), hasObs(o_2, o_1), hasSrc(o_2, txt)

In our approach, the general intuition is that all atomic observations, including source observations, have a source. To represent this constraint by excluding at the same time infinite chains of source observations, we introduce the notions of sourceChainRootIn, assertChainRootIn, and kb (where assertChainRootIn is a restriction of sourceChainRootIn to assertion chains rather than assertion/rejection chains). In principle, the source of a compound observation could be reduced to the compound of all the sources of its atomic parts. To simplify our framework we do not consider this aspect here.

[8]In the SWRL language, SameAs is the built-in construct to model identity between A-box individuals.
[9]For simplicity, we do not represent in (e6) the arguments of AuthorshipObs, which are however needed.

From an application perspective, users of the ontology do not need to care about populating sourceChainRootIn and assertChainRootIn since they are automatically built using the recursive SWRL rules (r3)-(r6) from the observations present in the knowledge base. Rule (r3) says that when source observation x has observation (i.e., asserts or rejects) y, then x is a source chain rooted in (starting from) y. Recursively, (r4) says that when source observation x has observation (i.e., asserts or rejects) y, which has a source chain rooted in v, then x is a (larger) source chain rooted in v. In the previous example, o_2, o_3 and o_4 are all source chains rooted in o_1; o_3 and o_4 are source chains rooted in o_2; and o_4 is a source chain rooted in o_3. Similarly for (r5) and (r6) in the case of assertChainRootIn.

r3 $\mathsf{SourceObs}(?x), \mathsf{hasObs}(?x, ?y) \rightarrow \mathsf{sourceChainRootIn}(?x, ?y)$

r4 $\mathsf{SourceObs}(?x), \mathsf{hasObs}(?x, ?y), \mathsf{sourceChainRootIn}(?y, ?v) \rightarrow \mathsf{sourceChainRootIn}(?x, ?v)$

r5 $\mathsf{AssertObs}(?x), \mathsf{hasObs}(?x, ?y) \rightarrow \mathsf{assertChainRootIn}(?x, ?y)$

r6 $\mathsf{AssertObs}(?x), \mathsf{hasObs}(?x, ?y), \mathsf{assertChainRootIn}(?y, ?v) \rightarrow \mathsf{assertChainRootIn}(?x, ?v)$

The individual kb is an instance of Text representing the knowledge base of the system itself. It is intended to document all the existing positions expressed in the analyzed texts, rather than to express its own viewpoint on the texts. The interpretative dimension of kb is therefore unavoidable and is explicitly represented by assertion and rejection observations with source kb.

We assume that all source chains end in kb; thus avoiding infinite source chains. This constraint is formally represented by (i) introducing the subclass KBAssertObs of AssertObs and the subclass KBRejectObs of RejectObs characterized via (a6) and (a7), respectively, i.e., these classes identify the source observations with source kb; (ii) introducing the subproperty kbSourceChainRootIn of sourceChainRootIn that is automatically populated starting from KBAssertObs, KBRejectObs, and sourceChainRootIn by means of the SWRL rules (r7)-(r10); and (iii) adding the axiom (a8) claiming that every atomic observation is the root of a source chain ending in the kb. Note that, by assuring that kb-observations are rooted in themselves, (r9) and (r10) are "technical artifices" that make (a8) operative. Finally, we assume that kb is "coherent" (see (r11)): given the disjointness of AssertObs and RejectObs, (r11) generates an inconsistency when kb asserts and rejects the same observation.

a6 $\mathsf{KBAssertObs} \equiv \mathsf{AssertObs} \sqcap \exists \mathsf{hasSrc}.\{kb\}$

a7 $\mathsf{KBRejectObs} \equiv \mathsf{RejectObs} \sqcap \exists \mathsf{hasSrc}.\{kb\}$

r7 $\mathsf{KBAssertObs}(?a) \rightarrow \mathsf{sourceChainRootIn}(?a, ?a)$

r8 $\mathsf{KBRejectObs}(?a) \rightarrow \mathsf{sourceChainRootIn}(?a, ?a)$

r9 $\mathsf{KBAssertObs}(?a), \mathsf{sourceChainRootIn}(?a, ?o) \rightarrow \mathsf{kbSourceChainRootIn}(?a, ?o)$

r10 $\mathsf{KBRejectObs}(?a), \mathsf{sourceChainRootIn}(?a, ?o) \rightarrow \mathsf{kbSourceChainRootIn}(?a, ?o)$

a8 $\mathsf{AtomicObs} \sqsubseteq \exists \mathsf{kbSourceChainRootIn}^{-}.(\mathsf{KBAssertObs} \sqcup \mathsf{KBRejectObs})$

r11 $\mathsf{hasObs}(?a, ?o), \mathsf{hasObs}(?r, ?o), \mathsf{hasSrc}(?a, kb), \mathsf{hasSrc}(?r, kb) \rightarrow \mathsf{SameAs}(?a, ?r)$

In practice, in order to block infinite regressions, for every observation introduced under Observation, users need to provide a chain of observations ending at kb. It is important to stress that in our approach kb is not intended as a direct source of information; it is rather a collector of observations coming from other sources. Formally, this means that there should be no observation having kb as a direct source, hence, at least a chain of length 2 is assumed (see examples in Section 4).

3.2. Analysis of the debates

Once we collect basic observations linked to specific observation languages, together with source and argumentation observations concerning them, we can analyze the status of some claims or sources (e.g., coherence, ambiguity, conflict) on the basis of explicit criteria. In the following, we mainly adapt principles considered in *assertion logics* [15], and *argumentation frameworks* [16, 14, 17]. The general strategy consists in introducing SWRL rules corresponding to these criteria to automatically populate some classes or object properties to represent the status of claims or sources.

Let us begin with the notion of *disputability* of observations: an observation o is *strongly disputable* when there are two sources directly accessible by kb, one asserting and one rejecting o. This notion can be captured through the class StronglyDisputableObs and the rule (r12), ensuring that when there are two kb assertions, one asserting that a source asserts o, and one asserting that a source rejects o, then o is strongly disputable. According to (r12), kb has *direct* access to the observations that o is asserted and rejected by some source. Strong incoherence can be weakened by allowing kb to have *indirect* access to such observations, i.e., they are reported to kb via a chain of assertions. To represent this, one can rely on kbAssertChainRootIn (a single rejection in a chain would prevent the transfer of information from the original source to kb) to introduce the rule (r13) for the new class DisputableObs.

r12 $\mathsf{KBAssertObs}(?x), \mathsf{KBAssertObs}(?y), \mathsf{hasObs}(?x, ?a), \mathsf{hasObs}(?y, ?r),$
$\mathsf{AssertObs}(?a), \mathsf{RejectObs}(?r), \mathsf{hasObs}(?a, ?o), \mathsf{hasObs}(?r, ?o) \rightarrow \mathsf{StronglyDisputableObs}(?o)$

r13 $\mathsf{kbAssertChainRootIn}(?x, ?a), \mathsf{kbAssertChainRootIn}(?y, ?r),$
$\mathsf{AssertObs}(?a), \mathsf{RejectObs}(?r), \mathsf{hasObs}(?a, ?o), \mathsf{hasObs}(?r, ?o) \rightarrow \mathsf{DisputableObs}(?o)$

A similar analysis can be done focusing on the *incoherence* of sources, i.e., when a source both asserts and rejects an observation. See the OWL file in the repository for the SWRL rules and the new classes necessary for the two versions of incoherence.

Following argumentation theory, we can also study some properties of the "network" of support and defeat observations. In particular, we can analyze compound observations collecting all the observations relevant for a given context by classifying them under some classes.[10] For example, a (compound) observation is *conflictual* when it has two parts one defeating the other. A strong notion of conflict is introduced through the rule (r14). Again, such notion can be weakened by considering assertions chains.

r14 $\mathsf{partOf}(?x, ?o), \mathsf{partOf}(?y, ?o), \mathsf{KBAssertObs}(?a), \mathsf{hasObs}(?a, ?d), \mathsf{DefeatObs}(?d),$
$\mathsf{hasDefeatingObs}(?d, ?x), \mathsf{hasDefeatedObs}(?d, ?y) \rightarrow \mathsf{StronglyConflictualObs}(?o)$

4. Case study: Observations in Literary Studies

In this section, we exemplify how the core module of our framework can be applied to literary studies, and notably how it can be used to model specific kinds of scholarly observations about the contents of literary texts. We partially reuse the case study presented in a first-order setting of our framework [8], whereas we now show how it can be modeled in OWL.

The case study consists of the representation of observations by different scholars concerning the final *novella* (tale) of Boccaccio's *Decameron*, namely the tale (X,10) of Griselda and Gualtieri. In short, the tale narrates the story of the nobleman Gualtieri who marries the poor Griselda. To test his wife's obedience, Gualtieri submits Griselda to increasingly cruel trials, such as letting her believe that their children need to be murdered. Griselda accepts everything he puts her through. In the end, Gualtieri reveals the truth and they live happily ever after. Due to its controversial plot, the critical debate around the tale is still open. We focus on how Vittore Branca and Michelangelo Picone analyzed the tale's (literary) characters, as well as how they conceived the relationships between the tale and other texts. For shortness, we shall report only some observations; the reader can refer to the available OWL file, as well as to [8] for an in-depth overview of the case study. From a methodological perspective, it is important to stress that the documented observations, as well as the formal structure of the observational vocabulary, have been selected and designed through close interaction with literary scholars.

Among his main theses, Branca characterizes Griselda by linking Boccaccio's work to biblical and hagiographic texts. In particular, he argues for the similarity between Griselda and the Virgin Mary, which is in turn supported by the observation that they are both patient figures. The theses by Picone follow a different direction, connecting Boccaccio's tale to the Medieval *Lai de Fresne*. In doing so, Picone investigates the relationships between the characters of Griselda and Fresne, rejecting the similarity between Griselda and Mary.

[10] Argumentation theory usually refers to sets of arguments rather than compound arguments.

To make sense of these observations in a modular extension of our ontology, we need to introduce at least the following individuals in the domain of quantification: the figures of Griselda ($griselda$) and Virgin Mary ($mary$), Branca's text *Boccaccio medievale* (bmd), Picone's text *Boccaccio e la codificazione della novella* (bcn), a collection of hagiographic texts (hag),[11] and the *Decameron*'s tale X,10 (tlx). At the class level, we introduce two classes of basic observations: (i) PatientObs for the observation of being a patient figure (a9); and (ii) ResemblanceObs for the observation of being resemblant to other domain entities (a10). As in previous examples, in the case of (a9), the axiom makes explicit the range of hasArgument to a specific class of the ontology, which is Figure in this case. We introduce the latter class only in a preliminary way for the sake of the example to model domain entities like Griselda and Virgin Mary; the modeling of (literary) characters deserves much more robust work to be done in a principled manner. In the case of (a10), hasResemblantArg and hasResembledArg are subrelations of hasArgument meant to distinguish between the two arguments of the observation.

a9 $\mathsf{PatientObs} \sqsubseteq \exists^{=1}\mathsf{hasArgument.Figure}$

a10 $\mathsf{ResemblanceObs} \sqsubseteq \exists^{=2}\mathsf{hasArgument}$,
$\mathsf{ResemblanceObs} \sqsubseteq \exists^{=1}\mathsf{hasResemblantArg}$, $\mathsf{ResemblanceObs} \sqsubseteq \exists^{=1}\mathsf{hasResembledArg}$

Let us assume that it is not controversial that Griselda is a figure; hence, we can introduce $\mathsf{Figure}(griselda)$ (analogously for the Virgin Mary). (e7) represents that, according to Branca's text bmd, Boccaccio's tale tlx asserts that Griselda is patient: o_1 is the observation of Griselda being patient; o_2 is the observation of tlx asserting o_1; o_3 is the observation of bmd asserting o_2 (i.e., o_3 is the assertion done in bmd that tlx asserts that Griselda is patient). Finally, o_4 is the assertion of o_3 in kb which is required to close the source chains. Similar observations are introduced for the characterization of the Virgin Mary with respect to the text *hag* (see the OWL file).

e7 $\mathsf{PatientObs}(o_1), \mathsf{hasArgument}(o_1, griselda), \mathsf{AssertObs}(o_2), \mathsf{hasObs}(o_2, o_1), \mathsf{hasSrc}(o_2, tlx)$,
$\mathsf{AssertObs}(o_3), \mathsf{hasObs}(o_3, o_2), \mathsf{hasSrc}(o_3, bmd), \mathsf{KBAssertObs}(o_4), \mathsf{hasObs}(o_4, o_3)$

According to (e8), o_9 is the observation of Griselda resembling the Virgin Mary; o_{10} is the assertion of o_9 expressed in bmd; and o_{11} is the required assertion of o_{10} in kb to close the source chain.

e8 $\mathsf{ResemblanceObs}(o_9), \mathsf{hasResemblantArg}(o_9, griselda), \mathsf{hasResembledArg}(o_9, mary)$,
$\mathsf{AssertObs}(o_{10}), \mathsf{hasObs}(o_{10}, o_9), \mathsf{hasSrc}(o_{10}, bmd), \mathsf{KBAssertObs}(o_{11}), \mathsf{hasObs}(o_{11}, o_{10})$

(e9) represents the argument put forward by Branca in favour of the resemblance between Griselda and the Virgin Mary: o_{13} is the supporting of o_9 (the resemblance of Griselda with the Virgin Mary) by o_{12}, where o_{12} is the compound of o_2 (*tlx* asserting Griselda being patient) with o_6 (*hag* asserting Virgin Mary being patient, see the OWL file);[12] o_{14} is the assertion of the supporting observation o_{13} in bmd (we omit here o_{15} as assertion of o_{14} in kb).

e9 $\mathsf{properPartOf}(o_2, o_{12}), \mathsf{properPartOf}(o_6, o_{12}), \mathsf{SupportObs}(o_{13}), \mathsf{hasSupportingObs}(o_{13}, o_{12})$,
$\mathsf{hasSupportedObs}(o_{13}, o_9), \mathsf{AssertObs}(o_{14}), \mathsf{hasObs}(o_{14}, o_{13}), \mathsf{hasSrc}(o_{14}, bmd)$

For lack of space, (e10) reports only the source observation of Picone's text bcn rejecting o_9, i.e., the resemblance of Griselda with the Virgin Mary (as for o_{14} in (e9), we omitted the assertion of o_{16} in kb).

e10 $\mathsf{RejectObs}(o_{16}), \mathsf{hasObs}(o_{16}, o_9), \mathsf{hasSrc}(o_{16}, bcn)$

Reasoning over the ontology and the data, by means of (r12) and (r13), o_9 is classified by both StronglyDisputableObs and DisputableObs, i.e. there is a disagreement about the observation that

[11]The introduction of hag is a simplification with respect to Branca, who talks of hagiographies without reference to specific texts. This might lead to the modeling of "vague" entities, which we leave to future work.

[12]The symmetry of resemblance is not assumed, which motivates the introduction of both hasResemblant and hasResembled as specializations of hasArgument. This is because, first of all, there are open discussions in cognitive science about the symmetry of general *similarity* relations; second, we do not commit to the fact that the resemblance Branca has in mind is symmetric, even though it is supported by the sharing of a common trait (being patient) among the two figures. Furthermore, symmetric relations raise a problem concerning observations, e.g.: is it necessary to distinguish 'Griselda resembling the Virgin Mary' from 'the Virgin Mary resembling Griselda', or is the single observation 'the resembling of Griselda and the Virgin Mary' enough? We leave this open to future work.

Griselda resembles the Virgin Mary. Depending on the application context and the represented data, our framework can support the automatic classification of individuals in various ways, e.g., to spot the presence of incoherent sources, conflictual observations, and so on.

5. Comparison with State of the Art

The representation of observations in the Digital Humanities has been attracting the interest of the community for a while. We now wish to compare our approach with some existing works.

The CRM Argumentation Model (CRMinf) [18] is an extension of the CIDOC-CRM ontology (CRM for shortness) to support the documentation of argumentation. The class *I1 Argumentation* plays a prominent role in CRMinf, where it is intended as an activity resulting in a belief, the latter having a proposition set as content (*what* the belief states), and a belief value (true, false, etc.). CRMinf also allows users to document some aspects concerning the logic of argumentation, e.g., that beliefs can be premises for other beliefs. It is important to recall that CRM ontologies exist as conceptual models, whose formal representations can be differently designed depending on their applications. At the current state, CMRinf exists as an RDFS specification.

The Historical Context Ontology (HiCO) [2] is an OWL ontology built to represent scholarly statements in the humanities.[13] In particular, HiCO is developed "for representing the *context* of a claim. [...] For instance, being created by somebody, or being created at a certain time [...]" (emphasis is ours; from the online documentation, see footnote 13). In this sense, the ontology focuses on the meta-data of claims and is not meant to model *what* is claimed. Notice that, even if not explicitly introduced in the present paper, our ontology can be extended to capture the temporal dimension associated with observations, e.g., by (i) importing existing resources such as the Time ontology[14]; and (ii) adding the object property hasTime (or one of its specializations, e.g., hasBeginning, hasEnd) to capture the relationship between observations and times. Unlike the timestamp-based approach in HiCO (realized by the use of the PROV-O data-property startedAtTime),[15] importing a theory based on the algebra of binary relations on intervals [19] would allow us to perform non-trivial reasoning about the chronological development of literary criticism.

To model claims' contents, some of the authors of HiCO have been developing an approach based on *named graphs*, see [20]. They distinguish between different sorts of RDF graphs to separate hypothetical statements, whose truth-value is neither true, nor false, from statements whose truth-values have been settled by scholars. The proposal is still in its preliminary development; at the current state, it is mainly developed for query-answering tasks, rather than for reasoning over data, e.g., to automatically detect possible conflicts between alternative claims. On this respect, the notion of "collapse graph" introduced in [20] to deal with the representation of the so-called "settled disputes", i.e., claims that have been put in doubt by someone, but for which eventually a consensus is reached, is paradigmatic and marks a major difference with our approach. While collapse graphs, and therefore the existence of settled claims in a given scenario, are the result of a modeler's choice, our system is powerful enough to reason about the existence of such claims. More specifically, the ontology can be extended to deduce that an observation is object of common agreement (i.e., is a settled claim) because all the available sources (or an authoritative selection of them) assert it. In addition, the notion of conflictuality discussed in Section 3.2 can be modified and extended to take into account the available background or commonsense knowledge; e.g., knowing that the date of birth of a person is unique would support the categorization of two different date-observations as conflictual.

To make a distinction between factual and hypothetical data in cultural heritage, Carriero et al. [21] adopt a pattern for *interpretation situations* based on previous work by Gangemi et al. [22]. These situations are used to represent the attribution of properties to domain entities. Sartini et al. [23] have extended this approach with application in art history.

[13]HiCO: https://marilenadaquino.github.io/hico/.
[14]Time Ontology: https://www.w3.org/TR/owl-time/.
[15]PROV-O: https://www.w3.org/TR/prov-o/.

Stressing a bit further some key differences between our approach and the ones just mentioned, from a conceptual standpoint, we do not take observations as mental entities, unlike CRMinf. As said in Section 2, they are pieces of information expressed through observational languages. This is an important departure point: our aim is to model the claims that scholars express in accessible sources, rather than their mental ideas. We also depart from pure technical approaches like the one in [21, 22], where (interpretation) situations are OWL patterns for n-ary relations. Our framework is grounded in foundational research to make an explicit difference between hypothetical and factual data (see Section 2). In addition, as an important difference with the state of art, it is possible to introduce identity criteria for observations, allowing for the identification of observations in an unambiguous way.

Another key aspect of our framework is its axiomatization, which makes it applicable in contexts where both query-answering and reasoning can be applied. From this perspective, we have relied on existing work in argumentation theory to represent some elements of the argumentation put forward by scholars to support or defeat claims. Moreover, we have introduced various SWRL rules to automatically analyze claims and their relations on the basis of transparently designed criteria, e.g., to check whether observations express conflictual claims.

6. Conclusion

In this paper we presented the core module of a Semantic Web ontology to represent and reason over observational data, collected through the design of (domain-specific) controlled observational languages. These latter need to be developed in close collaboration with domain experts to reflect as much as possible their critical stance on the observed entities. It is important to stress that to make sense of the variety of conceptual systems in the humanities, our approach allows for the development of alternative observational languages, possibly linked in what we call a library of observational languages, to represent different sorts of claims and their mutual relationships. We also introduced SWRL rules for the analysis of both observations and their sources, e.g., to check whether a single observation makes conflictual claims. Further rules can be added by extending our ontology and making it suitable to specific application contexts. Conceiving the framework with a modular architecture means to make it available for different domains. By collaborating with domain experts to test the ontology against their knowledge, we have also shown a preliminary exemplification of the use of our ontology in literary studies, giving a sample of its potential uses to represent scholarly claims. Further work is needed to refine our ontology with respect to a larger quantity and variety of data.

As a final remark, it is important to stress that our approach is compatible with research and application efforts in Computational Literary Studies (CLS) [9], where machine learning or natural language processing techniques are used to automatically identify patterns and extract information from large text corpora. For instance, our formal system explicitly models certain sorts of data as *observations* made by scholars in specific textual contexts. This can lead to novel platforms to store, share, interlink, and analyze observational data through different CLS techniques supporting reception and interpretation studies. In addition, CLS pipelines can be enriched by the adoption of observation languages that are formally axiomatized, hence, by symbolic reasoning methods leading to transparent and human-controllable results.

Acknowledgments

Sanfilippo and Tomazzoli's work is supported by the project *MITE – Make it explicit: Documenting interpretations of literary fictions with conceptual formal models* (2023-2025) funded by the European Union – Next Generation EU, Mission 4, Component 1, CUP B53D23028830001. Mosca's work is supported by the PNRR MUR project PE0000013-FAIR project *Future Artificial Intelligence Research* funded by the European Union – Next Generation EU.

References

[1] G. Barabucci, F. Tomasi, F. Vitali, Supporting complexity and conjectures in cultural heritage descriptions, in: Proceedings of the International Conference Collect and Connect: Archives and Collections in a Digital Age (COLCO 2020), CEUR workshop proceedings, volume 2810, 2021.

[2] M. Daquino, F. Tomasi, Historical context ontology (HiCO):A conceptual model for describing context information of cultural heritage objects, in: Metadata and Semantics Research: 9th Research Conference, 2015, Proceedings, Springer, 2015, pp. 424–436.

[3] M. Doerr, A. Kritsotaki, K. Boutsika, Factual argumentation—a core model for assertions making, Journal on Computing and Cultural Heritage (JOCCH) 3 (2011) 1–34.

[4] I. Willis, Reception, Routledge, 2017.

[5] A. Pichler, N. Reiter, From concepts to texts and back: Operationalization as a core activity of digital humanities, Journal of Cultural Analytics 7 (2022).

[6] K. Janowicz, A. Haller, S. J. Cox, D. Le Phuoc, M. Lefrançois, SOSA: A lightweight ontology for sensors, observations, samples, and actuators, Journal of Web Semantics 56 (2019) 1–10.

[7] C. Masolo, A. Botti Benevides, D. Porello, The interplay between models and observations, Applied Ontology 13 (2018) 41–71.

[8] E. M. Sanfilippo, A. Sotgiu, G. Tomazzoli, C. Masolo, D. Porello, R. Ferrario, Ontological modeling of scholarly statements: A case study in literary criticism, in: Formal Ontology in Information Systems (FOIS 2023), volume 377 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2023, pp. 349–363.

[9] H. O. Hatzel, H. Stiemer, C. Biemann, E. Gius, Machine learning in computational literary studies, it-Information Technology 65 (2023) 200–217.

[10] E. Gius, J. Jacke, The hermeneutic profit of annotation: On preventing and fostering disagreement in literary analysis, International Journal of Humanities and Arts Computing 11 (2017) 233–254.

[11] K. Mulligan, F. Correia, Facts, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Winter 2021 ed., Metaphysics Research Lab, Stanford University, 2021.

[12] M. McGrath, D. Frank, Propositions, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Winter 2020 ed., Metaphysics Research Lab, Stanford University, 2020.

[13] A. J. Cotnoir, A. C. Varzi, Mereology, Oxford University Press, 2021.

[14] C. Cayrol, M.-C. Lagasquie-Schiex, Bipolarity in argumentation graphs: Towards a better understanding, International Journal of Approximate Reasoning 54 (2013) 876–899.

[15] N. Rescher, Topics in philosophical logic, Springer Science & Business Media, 1968.

[16] P. M. Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, Artificial intelligence 77 (1995) 321–357.

[17] G. Brewka, Dynamic argument systems: A formal model of argumentation processes based on situation calculus, Journal of logic and computation 11 (2001) 257–282.

[18] M. Doerr, C.-E. Ore, P. Fafalios, Definition of the CRMinf. An Extension of CIDOC-CRM to support argumentation, Technical Report, CIDOC CRM-SIG, 2023.

[19] J. F. Allen, Towards a general theory of action and time, Artificial intelligence 23 (1984) 123–154.

[20] V. Pasqual, G. Manzano, E. Uzeir, F. Tomasi, F. Vitali, et al., Evaluation of expressing without asserting approaches in rdf. the case of conjectures., in: IRCDL, 2024, pp. 60–73.

[21] V. A. Carriero, A. Gangemi, M. L. Mancinelli, A. G. Nuzzolese, V. Presutti, C. Veninata, Pattern-based design applied to cultural heritage knowledge graphs, Semantic Web 12 (2021) 313–357.

[22] A. Gangemi, P. Mika, Understanding the semantic web through descriptions and situations, in: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer, 2003, pp. 689–706.

[23] B. Sartini, S. Baroncini, M. van Erp, F. Tomasi, A. Gangemi, Icon: An ontology for comprehensive artistic interpretations, ACM Journal on Computing and Cultural Heritage 16 (2023) 1–38.