ICBO: International Conference on Biomedical Ontology
July 24-26, 2009 ▪ Buffalo, New York

A Formal Ontology of Sequences

Robert Hoehndorf[1,2], Janet Kelso[2], Heinrich Herre[1]
[1]IMISE, University of Leipzig, Germany; [2]Department of Evolutionary Genetics, Max Planck Institute for Evolutionary Anthropology, Leipzig, Germany

Abstract

The Sequence Ontology is an OBO Foundry ontology that provides categories of sequences and sequence features that are applied to the annotation of genomes. To facilitate interoperability with other domain ontologies and to provide a foundation for automated inference, we provide here an axiom system for the Sequence and Junction categories in first- and second-order predicate logics.

Introduction

Biological sequences play a major role in genetics and bioinformatics research. They are important in the description of DNA, RNA and proteins. To describe sequences and their features semantically, the Sequence Ontology (SO)[2] was developed.

The SO distinguishes between sequence features, qualities of sequences, operations on sequences and sequence variants. A sequence feature is an extended or non-extended biological sequence. Extended sequence features are regions such as genes, intergenic regions or sequences of polypeptides. Non-extended sequences are called junctions – the boundaries between two extended sequences. Operations on sequences include insertions and deletions. Qualities of sequences include whether or not a sequence encodes a protein, whether a sequence acts enzymatically when transcribed, or whether the sequence is conserved. Although some formal definitions are available for the SO categories, most categories are defined using English.

Formal ontologies are intended to specify a conceptualization of a domain[5], and therefore provide the foundation for data and information integration and exchange. Definitions alone are insufficient to achieve this goal. Axioms are required to provide meaning for primitive, undefined categories. To formalize the basic categories used in the SO, several ontological questions about sequences must be answered, among them: What kind of entity is a biological sequence and how does it relate to categories in a top-level ontology? What are the properties of biological sequences? What relations are applicable to sequences? How do sequences relate to other kinds of entities, in particular to molecules, organisms or processes (of selection, mutation)?

Here we provide an axiom system for the SO's top-level categories. We use first- and second-order logics for this purpose. The axiom system is intended to serve as a foundation for the SO, and as a means to achieve interoperability between the SO and other domain ontologies through the provision of an explicit formalization of the basic categories and relations used in the context of sequences. For the construction of the axiom system, we employed the axiomatic method[8].

Method

We consider a formal ontology to be a specification of a conceptualization, i.e., a particular view on the world[5]. A formal ontology uses a vocabulary whose terms denote concepts and relations which refer to things in reality.

One method that is used to specify the meaning of a term is an explicit definition. An explicit definition for a relation or category P provides a sentence ϕ in which P does not occur, such that every occurrence of P can be replaced with ϕ.

When explaining the meanings of a set of terms through explicit definitions, other terms must be used to define the terms in the set, and in turn the meaning of these terms must be specified (without creating a circular definition). Therefore, specifying the meanings of terms solely through explicit definitions will either lead to an infinite regress or leave several terms unspecified. In the latter case, the meaning of all terms for which a definition is provided depends on the meaning of the terms without definition, therefore leaving the meaning of all terms in the ontology unspecified.

We call the terms that are not explicitly defined *primitive terms*. The meaning of all terms in the ontology depends on the meaning of these primitive terms: because non-primitive terms are introduced through explicit definitions, every sentence involving a non-primitive term can be replaced with a sentence containing only primitive terms.

The problem remains how the meaning of the primitive terms can be described formally. We may construct complex sentences containing only primitve terms. These sentences can be understood as descriptions of formal interrelations between the

primitive terms. Some of these sentences are chosen as axioms: they are accepted as being true within the domain under consideration. Such axioms provide restrictions on the interpretation of the primitive terms, and therefore on the terms defined using these primitive terms. For a formal theory, and therefore for a formal ontology, the axioms are the central component, because only they can give meaning to terms used in the theory.

Results

The theory of biological symbols and sequences that we propose here is intended to be compatible with the Sequence Ontology (SO)[2]. The SO uses two basic categories in the characterization of sequences, *Sequence* and *Junction*. Both can have attributes, i.e., properties. For example, a sequence may be a gene or a base, a junction an insertion site, and a sequence attribute enzymatic.

Sequences are linear entities and can come in two facets. Sequences can either have a start and an end point (such as an mRNA sequence), or form circles (such as the sequence of mitochondrial DNA). There are sequence atoms, which we call *Primitive biological symbols*. Primitive biological symbols have no proper sequence parts.

We introduce an important distinction that is currently neglected in the SO. The SO contains as its only basic category a sequence region, and employs an extensional mereological system on it. However, we will show that it is important to distinguish between a *sequence* and the *tokens* of a sequence. To illustrate the difference between a sequence and its token, consider all constituents (parts) of the sequences $ACAC$ and $CAAC$. The first sequence has as parts the sequences $ACAC$, ACA, CAC, AC, CA, A and C. The sequence $CAAC$ has as parts sequences $CAAC$, CAA, AAC, CA, AA, AC, A and C. It is remarkable that, although both sequences apparently have the same *length*, use the same primitive symbols (only A and C), and every primitive symbol occurs exactly twice in each sequence, $ACAC$ has seven sequences as part, while $CAAC$ has eight. This is due to the fact that there is *only one* sequence AC, which occurs in $ACAC$ *twice*. On the other hand, each *token* of $ACAC$ and of $CAAC$ will have ten parts.

The theory we propose here assumes that *Sequence*, *Molecule*, *Junction* and *Abstract sequence* are primitive categories. In particular, they are not defined, but characterized axiomatically. *Sequence* and *Junction* refer to representations of sequences such as those found in biological databases. Sequences have tokens which belong to the *Molecule* category. Molecules are material entities which are located in space and time. Instances of *Sequence* represent abstract, information bearing entities which are instances of *Abstract sequence*.

We make no commitment to a particular top-level ontology. The ontology of sequences presented here can stand on its own, and axioms are presented for all relations used in the theory. However, the foundation in a top-level ontology can benefit the interoperability between the presented ontology and other domain-specific ontologies, because the top-level ontology can provide a common interface for multiple domain ontologies.

The theory is based on these primitives: the categories *Seq* of biological sequences, *Jun* of junctions, *Mol* of molecules, *ASeq* of abstract sequences, and the relations **sPO** (sequence-part-of), **PO** (part-of), **aPO** (abstract-part-of), **binds**, **::** (token-of), **Rep** (representation), **between**, **end** and **conn**.

The first part consists of axioms that restrict the arguments of some of the relations[§]. Additionally, an axiom requiring all sequences to have only molecules as tokens is introduced.

$$sPO(x,y) \rightarrow Seq(x) \wedge Seq(y) \quad (1)$$

$$PO(x,y) \rightarrow Mol(x) \wedge Mol(y) \quad (2)$$

$$Seq(x) \rightarrow \forall y(y :: x \rightarrow Mol(y)) \quad (3)$$

Based on the relation **sPO**, we first define **sPPO** (proper sequence part) and the category of primitive biological symbols (*PBS*) as well as the **soverlap** and **sdisjoint** relations:

$$sPPO(x,y) \leftrightarrow sPO(x,y) \wedge x \neq y \quad (4)$$

$$PBS(x) \leftrightarrow Seq(x) \wedge \neg\exists y(sPPO(y,x)) \quad (5)$$

$$soverlap(x,y) \leftrightarrow \exists z(sPO(z,x) \wedge sPO(z,y)) \quad (6)$$

$$sdisjoint(x,y) \leftrightarrow \neg soverlap(x,y) \quad (7)$$

The relation sPO is a parthood relation that holds for sequences when one sequence contains the other as a sequence part. It satisfies reflexivity, transitivity and antisymmetry, and therefore forms a partial order.

$$sPO(x,y) \wedge sPO(y,z) \rightarrow sPO(x,z) \quad (8)$$

$$Seq(x) \rightarrow sPO(x,x) \quad (9)$$

$$sPO(x,y) \wedge sPO(y,x) \rightarrow x = y \quad (10)$$

[§]The remaining relations take defined categories as arguments and are introduced later.

The relation sPO also satisfies the strong supplementation principle, leading to an extensional mereology for sequences[6]:

$$\neg sPO(x,y) \rightarrow \exists z(sPO(z,x) \wedge sdisjoint(z,y)) \quad (11)$$

Sequences consist entirely of atoms with respect to the relation **sPO**. The following two axioms require that all sequences have atoms as part, and that they are constituted of only atoms:

$$Seq(x) \rightarrow \exists y(PBS(y) \wedge sPO(y,x)) \quad (12)$$

$$Seq(x) \rightarrow \neg\exists y(sPPO(y,x) \wedge \forall u(sPPO(u,x) \wedge PBS(u) \rightarrow sPO(u,y))) \quad (13)$$

Next, we restrict the arguments for the **between** and **end** relation, and introduce the relation **in** through an explicit definition.

$$between(j,p_1,p_2,s) \rightarrow Jun(j) \wedge PBS(p_1) \wedge PBS(p_2) \wedge Seq(s) \quad (14)$$

$$end(j,p,s) \rightarrow Jun(j) \wedge PBS(p) \wedge Seq(s) \quad (15)$$

$$conn(j_1,j_2) \rightarrow Jun(j_1) \wedge Jun(j_2) \quad (16)$$

$$in(j,s) \leftrightarrow \exists p_1,p_2(between(j,p_1,p_2,s)) \vee \exists p(end(j,p,s)) \quad (17)$$

$$Seq(x) \rightarrow \neg Jun(x) \quad (18)$$

$$Jun(x) \rightarrow \neg Seq(x) \quad (19)$$

The following set of axioms pertains to the **conn** relation of connectedness between junctions. It is used to represent the order of the sequence through an order of junctions.

$$conn(j_1,j_2) \rightarrow conn(j_2,j_1) \quad (20)$$

$$conn(j_1,j_2) \rightarrow j_1 \neq j_2 \quad (21)$$

$$in(j_1,s_1) \wedge in(j_2,s_2) \wedge \neg soverlap(s_1,s_2) \rightarrow \neg conn(j_1,j_2) \quad (22)$$

$$conn(j_1,j_2) \wedge in(j_1,s) \rightarrow in(j_2,s) \quad (23)$$

The axioms presented here are mostly first-order axioms and do not suffice to require connectedness of sequences. Instead, a second-order axiom is required to express the fact that sequences must be connected:

$$\forall s\forall P(\forall x(P(x) \leftrightarrow in(x,s)) \wedge \forall Q(\exists a Q(a) \wedge \forall x(Q(x) \rightarrow P(x)) \wedge \forall u,v(Q(u) \wedge conn(u,v) \rightarrow Q(v)) \rightarrow \forall x(P(x) \rightarrow Q(x)))) \quad (24)$$

The remaining axioms pertain to molecules, relate sequences to their tokens or the abstract sequences they represent. They can be found in[9] and in the machine implementation we provide with this paper.

A question that is not answered with these axioms is how sequences and junctions relate to categories commonly found in the top-level ontology. We believe these axioms to be compatible with most major top-level ontologies, in particular BFO[4], DOLCE[11] and GFO[7]. However, the foundation in these ontologies varies substantially.

In BFO, sequences and their junctions should be considered subcategories of *Generically dependent continuant*. A category A is generically dependent on the category B if for every instance of A, some instance of B must exist. In the framework of the BFO, sequences are generically dependent on their tokens. The difficulty that arises with such a view is that not every sequence is the sequence of a molecule. Therefore, the tokens must not be restricted to molecules which have the structure specified by the sequence, but must include textual and other digital representations as tokens of sequences. Junctions, on the other hand, always belong to a sequence and cannot exist without a sequence. Therefore, junctions should be considered as specifically dependent continuants which are dependent on sequences.

In DOLCE, the category *Abstract* is a sub-category of *Particular*. The main characteristic of abstract entities is that they do not have spatial nor temporal qualities, and they are not qualities themselves[3]. Sequences as well as junctions have this property, and the axioms we provide can be founded in the DOLCE ontology through an addition axioms:

$$Seq(x) \vee Jun(x) \rightarrow dolce:abstract(x) \quad (25)$$

Integration of our theory in GFO is similar to the scenario described in the DOLCE. Alternatively, GFO provides the category *Symbol structure*, to which both sequences and junctions can be assigned. Symbol structures are higher-order categories in the GFO, and the token-of relation $(::)$ falls together with the instantiation relation.

Implementation

We implemented the axiom system using the SPASS first-order theorem prover[12]. The implementation can be found on our project webpage[10]. Due to the restriction of SPASS to first-order logic, we could not implement the axiom requiring connectedness of sequences. This axioms necessitates the use of monadic second-order logics. Furthermore, a condition that sequences must be finite could not be implemented due to the restrictions of first order logic.

We employed the SPASS theorem prover on our axioms and attempted to prove the proposition $\phi \wedge \neg\phi$. If this logical contradiction can be derived from the axioms we provide, our axioms would be inconsistent. On the other hand, if our axioms are consistent, we expect SPASS to never terminate, because, in the general case, an automated consistency proof for first-order theories is impossible[1].

The SPASS theorem prover could not find a proof for the contradictory statement $\phi \wedge \neg\phi$ in three weeks time. However, this is merely an indication for consistency. A formal proof of the consistency, e.g., through the construction of a model, is subject to future work.

Conclusion

We provide an axioms system for sequences, junctions and molecules in predicate logics. Most of the axioms are available in first-order logic, although some require the use of second-order logic. The axiom system is intended to serve as a foundation of the Sequence Ontology's top-level categories *Sequence* and *Junction*. As a corollary from the axiom system, we introduced a class of sequence tokens, which we called *Molecule*. We find that in order to understand the category *Sequence*, it is necessary to consider the tokens of a sequence.

The axiom system we provide is not based on a particular top-level ontology, but is compatible with multiple top-level ontologies. We discuss how to include the theory of sequences in the BFO, DOLCE and GFO top-level ontologies. Depending on the top-level ontology used, sequences and junctions are considered different kinds of entities: generically dependent continuants in BFO, abstract individuals in DOLCE and higher-order categories in GFO.

This axiom system for sequences is – to the best of our knowledge – the first extensive axiom system for basic categories of an OBO Foundry ontology. With increasing demands for semantic interoperability and information flow between OBO Foundry ontologies, the importance of developing axiom systems likely will increase, because only axioms can provide a formal specification of a category's meaning, and therefore provide the foundation for automated inferences, information flow and integration. The new axioms are implemented for the SPASS theorem prover and can be downloaded from our website[10].

References

1. Church A. A note on the Entscheidungsproblem. *Journal of Symbolic Logic*, 1:40--41, 1936.
2. Eilbeck K, Lewis S, Mungall C, Yandell M, Stein L, Durbin R and Ashburner M. The Sequence Ontology: a tool for the unification of genome annotations. *Genome Biology*, 6(5), 2005.
3. Gangemi A. DOLCE Lite 397 OWL File. http://www.loa-cnr.it/Files/DLPOnts/DOLCE-Lite397.owl, 2003. OWL Annotation of *abstract* class.
4. Grenon P. BFO in a Nutshell: A Bi-categorial Axiomatization of BFO and Comparison with DOLCE. Technical report, University of Leipzig, Leipzig, 2003.
5. Guarino N. Formal Ontology and Information Systems. In *Formal Ontology in Information Systems: Proceedings of the First International Conference (FOIS'98)*, 1998.
6. Guizzardi H. *Ontological foundations for structural conceptual models*. PhD thesis, University of Twente, Enschede, The Netherlands, Enschede, 2005.
7. Herre H, Heller B, Burek P, Hoehndorf R, Loebe F and Michalek H. General Formal Ontology (GFO) – A Foundational Ontology Integrating Objects and Processes. Onto-Med Report, Research Group Ontologies in Medicine, Institute of Medical Informatics, Statistics and Epidemiology, University of Leipzig, Leipzig, 2006.
8. David Hilbert. Axiomatisches Denken. *Mathematische Annalen*, 78:405--415, 1918.
9. Hoehndorf R. *Basic considerations for improving interoperability between ontology-based biological information systems*. PhD thesis, University of Leipzig, 2009.
10. Hoehndorf R. Formal Ontology of Sequences. http://bioonto.de/pmwiki.php/Main/FormalOntologyOfSequences, 2009.
11. Masolo C, Borgo S, Gangemi A, Guarino N and Oltramari A. WonderWeb Deliverable D18: Ontology Library (final). Technical report, Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy, 2003.
12. Weidenbach C, Brahm U, Hillenbr T, Keen E and Theobald C. SPASS version 2.0. *In Proc. CADE-18*, pages 275--279, 2002. Springer.