1

An Ontological Approach to Canonical Typology: Laying the foundations for e-Linguistics

SCOTT FARRAR

Abstract

With the rise of the Web as an important medium for displaying and curating typological data, the next generation of linguists will have access to thousands of data sets on many of the world’s remaining languages. It is not out of the question that many or most of the grammars ever published and possibly many sets of field notes will be digitised and stored in a Web accessible format (cf. Google Book Search). As more and more resources are digitised, there is an ever-increasing need for interoperability of data and transparency regarding the intended meaning of annotation elements. Standardisation of terminology is certainly one approach; however, there are more basic challenges, including a clear articulation of a conceptual framework for specific areas of linguistics including, among many others, canonical typology. Put another way, there is currently no broadly accepted logic or computer language that would allow arbitrary descriptive data to be automatically published, searched, and visualised, i.e., no possibility of an e-linguistics. In this paper, we explore laying the foundations for such an enterprise by drawing on recent and emerging Web technologies, as well as from applied formal ontology. In particular, we will address work within the GOLD Community of Practice whose aim is the creation of a General Ontology for Linguistic Description. We will argue that the resulting conceptual framework can be employed to solve some basic issues concerning an implementation of canonical typology. An OWL-DL implementation of GOLD is presented with relevant examples for canonical typology.

1.1 Introduction

Canonical typology provides a conceptual framework for comparing languages by establishing canonical points for various linguistic dimensions. The canonical approach allows us to more precisely describe how one linguistic phenomenon differs from another by comparing them to a central, widely understood canon. It
is from this starting point that a possible implementation emerges, an enterprise that we will refer to as **ontology-driven canonical typology**. The other contributions to this volume serve to situate various approaches to canonicity. The ontological approach, however, needs some explanation.

First, the main goal of this enterprise is to establish a long-term, shared knowledge base of our scientific ideas, a knowledge base that is not only meant for human consumption, but one whose primary purpose is for machine processing. With the ever increasing mass of descriptive data, machine processing can facilitate rapid new discoveries and allow for the verification of old ones. Even the most skilled of linguists require many hours of research to analyse a single descriptive grammar and even more to place that description within the context of linguistic theory. The process is not scalable when hundreds or, for that matter, even dozens of grammars are in question. A solution that has served other fields such as genetics and medicine is to automate using technology from applied formal ontology and information science. For people, multiple views and local control of data are important. For machines, interoperability of databases and precise formal definitions are what matter. To achieve these goals, we as a field must take advantage of emerging technologies to encode, store and process descriptive data. We should provide a straightforward pathway for data providers to publish their data for other linguists and machines. Likewise, we should also facilitate the smartest search possible over the data. So, with the Web as a medium, the enterprise of using computers as the primary means to publish, search, and visualise descriptive data is referred to in this chapter as **e-linguistics**. The goal is then to situate an implementation of canonical typology within the e-linguistics effort.

Such an approach is not really new or radical. The subfield of corpus linguistics or any integrative data-driven approach to linguistics, e.g., Haspelmath *et al.* (2005), could be construed as an attempt at e-linguistics. More broadly we situate e-linguistics under the rubric of e-humanities, thus inheriting all the associated challenges, and potential rewards. E-humanities is inspired by the U.S. National Science Foundation’s Cyberinfrastructure Program and the e-Science movement in Europe and elsewhere. It concerns automated data analysis, text/media integration (interoperability), computational modelling, as well as other activities.

There are specific challenges that face e-linguistics. The most serious of these are a lack of infrastructure and the prevalence of ill-defined data types. In terms of infrastructure, there are few widely accepted field-wide standards, though the IPA, Unicode, and possibly ISO/Ethnologue language codes are exceptions. And, there are certainly widely accepted traditional practises, such as the format of interlinear glossed text, codified in the Leipzig Glossing Rules. In terms of data types, it is not a simple case of arguing for one data type or another, for example, tree structures instead of feature structures. In fact, the lack of accepted universal data types is more related to another issue: linguists do not usually agree on a core conceptualisation of linguistics (e.g., the basic inventory of feature types...
1.2 Background

As background to the e-linguistics approach, we present an overview of the key underlying frameworks and technologies. First, there is the knowledge engineering approach and the general use of ontologies to represent the domain of descriptive linguistics. Motivations for using an ontological approach have been given elsewhere, for instance in Farrar and Langendoen (2003) and Farrar (2007). However, a summary will be given here. After a brief introduction to knowledge engineering and ontologies, we give a detailed discussion of the main formalism to be used throughout the paper, namely that of Description Logic. The use of DL notation in this chapter is justified for reasons of brevity, as the XML syntax for OWL-DL is too verbose to present in running text. We then present a methodology whereby a knowledge base can be instantiated by using a Description Logic.
1.2.1 Knowledge engineering and ontologies

We define knowledge engineering as the task of representing the knowledge of a particular domain in a machine readable format. For a particular knowledge engineering task, the formal language used to represent the knowledge often has far-reaching effects as to what kinds of domain knowledge can be captured by the representation. Furthermore, the product of knowledge engineering, the knowledge base, can be used in conjunction with automated reasoning tools to produce new knowledge, to prove the consistency of existing knowledge, and to enhance search within the knowledge base. The central assumptions in a knowledge base are captured in the ontological theory, or the set of statements that make up the essential knowledge of the domain, in other words, the knowledge that must always hold if the theory is to be coherent. We refer to such sets of statements as simply the ontology.

The statements included in the ontology hold according to a particular conceptualisation of the domain. A conceptualisation is an abstract, simplified view of the world being modelled (Gruber, 1993). Due to the complexity of any real-world domain such as descriptive linguistics, a conceptualisation is necessarily a simplified approximation of reality. Still, that the conceptualisation is approximate does not preclude it from being useful. A major issue in the modelling of the linguistics domain is that various linguistic theories adopt different and often incompatible conceptualisations. In fact, from the standpoint of the ontologist, the aim of science may be cast as the search for the ideal conceptualisation. But admittedly, what is meant by “ideal” can vary according to the task at hand. For our task, we require only that our conceptualisation be rich enough to account for canonical linguistic concepts and for the differences expressed in various linguistic descriptions. The nature of this task relaxes some of the requirements on the ontological theory. An ontology for all of linguistics is, at this point, unachievable and would require deep consensus as to how linguistics, in fact how language, is conceptualised.

1.2.2 Description logic

The task of ontology building requires the use of logic as a means of axiomatisation. First-order logic (FOL), for instance, is one well understood language for this task and is often employed, in one form or another, for this purpose; see the ontologies of SUMO (Niles and Pease, 2001) and DOLCE (Masolo et al., 2003). An alternative to FOL in the design of knowledge-based systems is the class of logics known collectively as Description Logic (DL) (Baader et al., 2003). A DL is a less expressive, but highly structured fragment of first-order logic. Using a DL buys improved computational tractability but at the cost of expressivity. This means that algorithms for working with DLs will be fast, but that expressing certain concepts in a DL might not be possible. This section gives an introduction to this class of logics by discussing some of the key properties of DLs illustrated
by examples in typical DL notation. Furthermore, we limit our discussion here to the linguistics domain.

A Description Logic is a formal logic in the strict sense. That is, it has a formal syntax which specifies how to construct well-formed sentences and a formal semantics which relates those sentences to a model. A Description Logic, as with all formal logics, has an associated proof theory, or a system of how certain entailments follow from a set of sentences. The focus of this section is mainly on the syntactic operations of Description Logic, but supplemented with an informal discussion of semantics. For a full account of the semantics of Description Logic, see Baader et al. (2003).

Whereas the predicates in FOL have equal ontological status, those in a DL come in two sorts: concepts and roles. Concepts in a DL are represented as unary predicates, while DL roles are represented as binary predicates. What are referred to as constants in a first-order logic are referred to as individuals in Description Logic. Intuitively a concept represents a category or kind in the domain being modelled. A concept is a universal notion and can be instantiated by individuals. For example, Contemporary Standard Russian is an individual in the domain and an instance of the concept language variety. The relation of instantiation holds between concepts and individuals, making an individual an instance of some concept (Nardi and Brachman, 2003). Individuals are disjoint from concepts and cannot be instantiated or related by the subsumption relation. A role is a binary relation between individuals. Description logic by definition has only binary relations and, thus, relations of higher arity (e.g. ternary relations) are disallowed, one example of the limits of expressibility. The terms concept and role show that the origins of Description Logic lie in the early work on knowledge representation, particularly frame-based languages (Baader et al., 2003). In such languages, information is gathered together into structures called frames (structured objects), each particular type of which admits a specified set of possible attributes related by slots (roles).

The terms concept, individual, and role are particular to the body of literature concerning Description Logics. More general works in ontology and knowledge engineering use class, instance, and (binary) predicate or relation, instead of the DL-specific terms. In the language of OWL-DL, the term property is—confusingly—used in place of binary relation. Though the current work is meant to guide the reader in constructing OWL-DL ontologies, we use standard DL terminology throughout, mainly for the sake of consistency since logical formulae are given in DL notation. Thus, we will use concept, individual, and role.

Within a Description Logic system, concepts and roles are separated from individuals by partitioning the knowledge base into a TBox (short for terminology box) and an ABox (short for assertion box, in the sense that assertions are made about a given terminology). The TBox consists of axioms and statements about the domain in general in the form of logical sentences, while the ABox consists of facts about individuals. A DL knowledge base $KB$ may be defined
minimally as a TBox $T$ and an ABox $A$, i.e. $KB = \{T, A\}$, where $T$ is the union of the set of concepts with the set of roles in the domain, and $A$ is the set of individuals in the domain. Furthermore, the TBox also contains various axioms relating to concepts and roles, while the ABox contains statements relating to individuals, mostly in the form of expressions showing particular relations between individuals.

Description logic can be used to represent much more than just basic concepts and individuals. Complex, non-atomic concepts can be specified through logical statements. Statements in a DL differ considerably from those in standard FOL. Moreover, statements in a DL are expressed using class and role names, but with no variables. Thus, statement (1.1) gives an expression in a DL.

\[
\text{InflectedUnit} \equiv \text{GrammarUnit} \sqcap \exists \text{hasConstituent}.\text{InflectionalUnit} \quad (1.1)
\]

This can be glossed as: “The class InflectedUnit is defined as the intersection of the class of GrammarUnit and any class having at least one hasConstituent role whose value is restricted to the class InflectionalUnit.” Statements in DL are therefore formulae containing predicates, technically with one free-variable, but omitted in the syntax. Predicates in a DL represent concepts and roles. Concepts are either atomic, i.e. those identified by name and may or may not be defined,\(^1\) or complex, i.e. those derived from atomic concepts using a set of constructors. The supported concept and role constructors in a particular DL determine its expressive power (Horrocks et al., 2003, p. 6). Thus, the constructors are used to derive well-formed formulae. In the following we have listed some examples of very common constructors in DLs with notes about their respective semantics and how they could be used in an ontology for linguistics. Furthermore, these constructors and others are used to create right-hand side expressions that define anonymous concepts. Any of the expressions below could be placed with a named concept on the left and related with either $\equiv$ or $\sqsubseteq$.

**conjunction ($\sqcap$):**

\[
\text{AfricanLanguage} \sqcap \text{EndangeredLanguage} \quad (1.2)
\]

Statement (1.2) can be glossed as “those individuals which are shared between the concepts AfricanLanguage and EndangeredLanguage”. Conjunction is interpreted as the intersection of sets of individuals.

**disjunction ($\sqcup$):**

\[1\] Concepts that are never defined are referred to as primitive concepts.
1.2 Background

TenseFeature ⊔ AspectFeature \hspace{1cm} (1.3)

Statement (1.3) can be glossed as “the individuals that either belong to the concept TenseFeature or AspectFeature”. Disjunction is interpreted as union of sets of individuals.

negation (¬):

¬PhonologicalUnit \hspace{1cm} (1.4)

Statement (1.4) can be glossed as “the set of all individuals that are not instances of PhonologicalUnit”. Negation is interpreted as the complement of sets of individuals.

existential quantifier (∃):

∃ hasPart.GrammarUnit \hspace{1cm} (1.5)

Statement (1.5) can be glossed as “the set of individuals each member of which has some member of the set GrammarUnit as its part.” Two points should made clear. First, each member of the set must be related to at least one GrammarUnit by the hasPart role. Thus, the existential in Description Logic guarantees the existence of certain relationships. And second, the statement does not limit entities other than members of GrammarUnit from being parts. If individual IND is in the set given in (1.5), then the statement hasPart(IND, MORPH) where MORPH is a instance of Morpheme, is perfectly acceptable. The fact that other entities could be members of GrammarUnit is because of the open-world assumption built into the DL, namely that the domain is not assumed to be complete unless explicitly stated. Finally, the following serves to compare a simple DL formula (1.6) with the equivalent in standard FOL (1.7). Note that R stands for some role and C for some concept.

∃ R.C \hspace{1cm} (1.6)

\{x|∃y R(x, y) ∧ C(y)\} \hspace{1cm} (1.7)

Thus, at least one y is guaranteed to exists and is must be a member of C.

universal quantification (∀):

Universal quantification in Description Logic behaves in a similar fashion as the existential, but instead of the at-least-one criterion, members of the described set
must be related to only individuals of the concept given on the right, as in (1.8):

$$\forall \text{hasFeature.MorphosyntacticFeature}$$  \hspace{1cm} (1.8)

Statement (1.8) can be glossed as “the set of individuals whose features are only individuals of MorphosyntacticFeature”. Universal quantification requires all roles of some concept to be value-restricted by concepts of a certain type. Universal quantification does not, however, ensure that there will be a role that satisfies the condition, but if there are such roles, their ranges have to be restricted to the given type. Again, the following serves to compare a simple DL formula (1.9) with the equivalent in standard FOL (1.10).

$$\forall R.C$$  \hspace{1cm} (1.9)

$$\{x | \forall y \, R(x, y) \rightarrow C(y)\}$$  \hspace{1cm} (1.10)

1.2.3 Linguistic modelling with GOLD

Whereas Description Logic provides the formalism for representing an ontological theory, the General Ontology for Linguistic Description (GOLD) is an ontological theory for descriptive linguistics. Motivations for such an ontological approach have been given elsewhere Farrar and Langendoen (2003) and Farrar (2007), and a methodology for creating such ontologies is given in Farrar and Langendoen (2009). Minimally, an ontological theory specifies the entities of interest for a given domain. Those entities include classes and their instances along with the relations that hold among those instances. Light-weight ontologies stop there, by providing an enumeration of the classes and a limited number of relations, usually enough to arrange the classes according to a taxonomy. A catalogue of morphosyntactic features, perhaps with the subsumption relation, would be considered a light-weight ontology according to the terminology used here, though some do not refer to light-weight ontologies as ontologies at all. A more comprehensive formal ontology places many more restrictions on the entities in the domain and can serve to facilitate automated reasoning, for instance in deciding how close a certain grammatical system is to a canonical reference point.

The most important task in creating any ontology is to enumerate the entities found in the domain. If the inventory is ad hoc or incomplete, then the resulting ontology will not be an accurate conceptualisation. The key is to establish a rigid foundation such that later additions will not create problems for the overall theory. We refer to such a foundation as the upper ontology. For descriptive linguistics such an upper ontology contains the fundamental knowledge of structure, form, and meaning, that which is usually shared among a variety of linguists and across the subfields. This, ideally, would include general knowledge that applies to any language or theoretical framework. Examples of general knowledge of this sort are given below:
1.2 Background

- A verb is a part of speech.
- A verb can assign case.
- Gender can be semantically grounded.
- Linguistic signs have a semantic component.

This kind of declarative knowledge is typical of that represented in an ontology in knowledge-based systems. The ontology provides the means of formalising such expressions and defining them in a larger conceptual framework. For example, it provides the means of specifying how a spoken linguistic expression is related to the printed form of a writing system, or how the semantics of Tense is characterised in terms of a temporal calculus.

Of the most fundamental entities that occur in the linguistics domain are the linguistic expressions themselves. The basic entities here are: OrthographicExpression, SpokenExpression and SignedExpression. An OrthographicExpression has a physical form and is a special kind of symbolic string. The various modes of expression can have physical parts, such as OrthographicWord, or can be grouped to form a larger whole, such as OrthographicSentence.

Other than such concepts that are physical in nature, those occupying time and space, there are the abstract concepts such as the traditional units of linguistic analysis. As presented in Farrar (2007), these three types of entities are unified under the concept of LinguisticUnit. A linguistic unit is any element of linguistic analysis usually defined according to a particular level or stratum: form, meaning and grammatical structure. Language is multi-stratal (Halliday and Matthiessen, 2004, p. 24), because it can be analysed and described from a variety of points of view: in terms of form (or shape), meaning, or grammatical structure. For descriptive purposes at least, the separation of these different kinds of entities into various strata is necessary, because it is then possible to focus only on one stratum in an analysis, as is often done in descriptive linguistics. For example, consider the single bit of descriptive data, the Hungarian word emberek, meaning ‘people’. What we see printed on the page is a single orthographic representation. But assuming a linguistic analysis of emberek, we can talk about the existence of several different entities, all represented by the orthographic representation on the page. At the level of syntax, emberek represents a SyntacticWord. In terms of the phonology, it is a PhonologicalWord. And as for meaning, emberek signifies a meaning specific to Hungarian, glossed in English as ‘more than one person’.

Next, there are the entities that relate the fundamental units to one another. For instance, two expressions can be related via precedence in time and/or space, but also via dominance relations as in grammatical structure. The mereology of such units is a necessary component in the ontology, that is, how units are composed of other units. Consider the example of sound structure. There are the basic phonological entities, in general, PhonologicalUnit, subsuming the concept of Phoneme, PhonologicalWord, etc. An instance of PhonologicalWord is composed of instances of Syllable, just as instances of Syllable may be composed
of instances of Mora. Each level of linguistic analysis has its own unit types and theory, in short, its own mereology.

Finally, there are the various structuring devices used in linguistic analysis. In general, we refer to these as linguistic data types. There are several fundamental types, including Lexicon, GlossedText, PhonologicalParadigm, FeatureStructure, StructuralDescription, etc. Each of these data types has its own mereology, e.g. FeatureStructure which is the pairing of a feature name and a feature value.

1.2.3.1 The linguistic sign  The most fundamental modelling decisions in the upper ontology concern the linguistic sign. This final section is devoted to laying out some of these assumptions, as they will become relevant in the discussion of actual data in Section 1.5. Consider that in linguistic descriptions, as well as in ordinary speech, linguists do not generally recognise all three levels of analysis at the same time. Thus, when fish is used as an example, or described, a linguist might say “fish is a syntactic word that can take a possessive morpheme” or “fish is a mono-syllabic (phonological) word”. In terms of semantics, “fish is an animal” or “fish means or refers to a kind of animal” would be a common way to describe the meaning. How linguistic concepts are expressed in everyday speech is perhaps instructive for ontology. In particular, linguists use the form of a sign to refer to its various facets: formal, semantic or grammatical.

In terms of using Description Logic to represent ontology, a similar situation arises with objects and the various roles which they can assume. Consider, for example, a particular person BOB who is an instance of the concept MaleHuman. BOB may also be an instance of other concepts simultaneously such Teacher, Father, and Juror. Likewise, Bob’s yacht, YACHT1, can be modelled as an instance of Yacht, MortgagedProperty and ValuableObject.

The linguistic sign can be approached in the same way. What is represented by the sign fish is both a phonological word and a syntactic word. Because Description Logic allows for a single individual to belong to multiple concepts, we can simply declare the individual sign once FISH1, and then assert that it belongs to whatever types are relevant. Thus:

\[
\text{SyntacticWord(FISH1)} \\
\text{PhonologicalWord(FISH1)}
\]

But in order to preserve that FISH1 is a linguistic sign, SyntacticWord and PhonologicalWord are asserted to be subclasses of LinguisticSign, as in:

\[
\text{SyntacticWord} \sqsubseteq \text{LinguisticSign} \\
\text{PhonologicalWord} \sqsubseteq \text{LinguisticSign}
\]

A sign is distinct from its material form. The printed word fish is neither a phonological or syntactic unit. It is a physical object, defined as an entities
1.3 Canonical Typology

In this section we compare the ontology-driven method to the canonical approach discussed throughout the current volume. As the remaining sections will show, the ontology-driven method for linguistic description already resembles canonical typology and can be easily adapted to formalise key parts of the canonical approach. We first examine the idea that a conceptualisation can be regarded as a multidimensional space of sorts and then look at a specific ontological implementation of conceptual spaces. Finally, we turn to how individual canons can be compared to specific assertions found in the ontology.

1.3.1 Conceptualisation as canonical space

Consider the chapter by Greville Corbett where a canonical typology is characterised by a multidimensional theoretical space of possibilities. The space is a metaphor for the dimension by which various linguistic phenomena differ. In terms of ontology, the conceptualisation itself with its narrowly defined classes, instances and assertions is already a “space” of sorts, in that no other members
are allowed. The ontology is, therefore, a limiting space. As an example, consider a treatment of linguistic features. The issue concerns canonical knowledge versus specific knowledge of particular languages. When linguists refer to some notion like the hodiernal past in Nen, they are referring to the hodiernal past in the context of the Nen tense system. No other language variety has the Nen hodiernal past as one of its tenses, yet there exists languages (e.g., Cocama-Cocamilla) with tenses that behave similarly to the Nen hodiernal.

(1) mè nā nifũ sámũ hé bûnã nũmũ
   1SG HOD.PST parcel put:H LOC bed under
   ‘I have put the parcel under the bed.’


(2) Ritama- ca tus- ui
town- to go- HOD.PST
   ‘I went to town today.’

   Cocama-Cocamilla (ISO 696-3: cod) (Bybee et al., 1994, p. 98)

What is being proposed then is a canonical entity (a class) for each individual feature value type. Thus, there is the class HodiernalPast. A description of Nen would need to refer to an individual NenHP from that class:

\[
\text{HodiernalPast}(\text{NenHP})
\]  

(1.15)

Thus, though both occurrences of this feature seem to have the same label, \text{HOD.PST}, there is no identity. The alternate treatment where the \text{HOD.PST} refers to the same individual across languages would correspond to something like a cross-linguistic category, or gram type (see Bybee and Dahl 1989).

1.3.2 Conceptual vs. canonical spaces

Next, consider the metaphor of the canonical space as used throughout the current volume. There have been other formalisations of spaces using formal ontology, namely the treatment of qualities in DOLCE \(^2\) (Masolo et al., 2003). DOLCE uses the notion of a conceptual space to model qualities. Qualities are used to model the observable characteristics of entities such as physical objects and events. For instance, colour, shape and size are all modelled as qualities, as they can only exist by virtue of the physical entities that bear them. What is significant about DOLCE’s treatment is that there is a separation of actual qualities from their values, called qualia (Masolo et al., 2003, p. 17), much in the same

\(^2\) ‘DOLCE’ stands for Descriptive Ontology for Linguistic and Cognitive Engineering.
way that linguistic features are treated differently from their values. The relationship of a quality bearer to a quality and to its corresponding quale is shown in Figure 1.1. Inspired from Gärdenfors’ work on ‘conceptual spaces’, qualia in

DOLCE are modelled as subregions in particular quality regions. The notion of a region should not be taken literally to mean spatial region, but an abstract conceptual region (Gärdenfors, 2000). This is fitting, since DOLCE is a conceptually oriented ontology, or one that attempts to account for how conceptual systems are organised. Conceptual regions are, by virtue of the axioms of mereology and topology, well behaved and analogous to bona fide spatial regions. In fact, positions in real space can be modelled as special kinds of qualities (Bateman and Farrar, 2004). Quality regions can be modelled with $n$ dimensions, where $n > 0$. The subregions within a quality region can be $n$-dimensional, where $n \geq 0$. Furthermore, the framework ensures that individuals, particulars instances of classes, may not share the same individual qualities. But obviously, the intuition is that entities can share qualities, for example, the weight of two physical objects may be identical. In DOLCE’s framework, qualities share similar qualia, by virtue of the QT and QT relations (as shown in the figure).

The difference between a Quality and a point in quality space may be explained by reference to a distinction that is observable in natural language, given in (3).

(3) (a) This rose is red.

 (b) The colour of this rose is red.

 (c) The colour of this rose changed from red to brown.
The word *red* in (3a) is used as an adjective and refers to a particular instance of Quality. This is the colour that the particular rose described has. This quality of colour is uniquely that of this particular rose. In contrast, the *red* of (3b) is being used as a nominal rather than an adjective refers instead to an instance of a colour quale, or position in abstract colour space, subsumed under the concept *Region*. That this is necessary is argued by the existence of statements such as that of (3c); here it makes little sense to say that it is the colour ‘red’ itself that is changing from red to brown. It is only the particular instantiated colour of the rose that is taking up different colour values within the abstract colour region. We need, then, to keep these two layers of the account separate: the instantiated quality and the quale that it takes as a value.

At first such an account of qualities may seem overkill. But the notion of a quality region—or more precisely the separation of qualities from their values—provides a way to compare the individual qualities and to place them within a system, *something that could turn out to be quite useful in an ontological account of canonical spaces*. For instance, once canonical spaces are formalised, the task is then to determine “how particular linguistic objects differ in terms of their proximity to a point of convergence” (see Dunstan Brown’s chapter, this volume). Furthermore, it should be noted that DOLCE’s treatment of qualities differs from those found in other formal ontologies. SUMO (Niles and Pease, 2001), for instance, has the class *Attribute* whose instances—like in DOLCE—inhere to other entities via various relations such as ‘colour’ (cf. *qt* in DOLCE). Given the current SUMO treatment, however, there is no straight-forward way to express that *red* and *orange* are more similar than, say, *blue* and *orange*. In short, there is no way to express relationships among attributes. The benefit of using qualities and qualia to model observable characteristics of things is that once a quality space has been formalised, values in that space can be related to one another. Furthermore, a given quality can have values in more than quality space, depending the measuring system employed. This means that relations can be established across different quality spaces. Concerning linguistic features, this is exactly what is required. And, the notion of a quality region for linguistic features is exactly the mechanism needed for describing the feature systems of individual languages.

1.3.3 Canons

Turning to particular canons then, consider that “[the] canonical approach relies on establishing indisputable definitions which, at least in theory, are universally applicable and should not cause disagreement among linguists working in different theoretical frameworks” (see Nikolaeva’s chapter, this volume). For instance, these are sample canons that relate to finiteness:

C-1 tense marking > no tense marking
C-2 subject agreement > no subject agreement
Ontologies, GOLD in particular, establish definitions that are broadly applicable, but should be treated as defeasible. Such statements (assertions expressed using the logical machinery) can in principle be violated, provided that a reasoner (e.g., a theorem prover) is not used. This suggests one way of measuring deviation from the canon: determine how many assertions are violated by the actual data description and compare to the canon. But accommodating contradictions within an ontology may not be the best solution. There is another option, that of partitioning the ontological framework into upper- and sub-ontologies. For this, we turn to the notion of a Community of Practice Extension, or COPE for short.

1.4 COPEs

In terms of an ontological approach to canonical typology, the task is one of striking a balance between two opposing endeavours. The first is describing particular data from individual language varieties, a task that requires local knowledge. The second is adhering to the established canons (even conventions) that many linguists assume in scholarly work, a task that requires canonical knowledge. In other words, whereas a description of particular language data allows the describer to provide local knowledge of observed phenomena, any local description should adhere to the norms of the field. The extent to which this is possible is largely to be determined by the current project. In the simplest case, a local description requires a concept that is already available in established linguistic knowledge, for instance some commonly assumed linguistic feature.

But, there are two cases where established knowledge does not provide an accurate base for a descriptive work. First, if the local phenomenon expands established knowledge. For instance, a new type of linguistic feature is posited (e.g., a new kind of tense) or a new structural category is posited (e.g., a new type syntactic category). Secondly, there is the case where local knowledge contradicts established knowledge. This latter case would occur if the established knowledge is simply inaccurate or, more likely, the established knowledge is constrained (by assumptions of the field) in a way that is too rigid for all individual cases. An example of this would be something like the following statement: “accusative case only occurs on nouns”. Such a statement would need to be revised if a described variety were found to have accusative case on another syntactic category, say on adjectives. The last case resembles problems addressed by the canonical approach to typology where local knowledge somehow overrides canonical assertions.

Cast in the ontological framework, both cases may be handled by Community of Practice Extensions (COPEs) as described in Farrar and Lewis (2007). A COPE is simply an extension of the upper ontology, used in cases where the
knowledge required to describe particular data is lacking, or contradicts established knowledge. For instance, the upper ontology may only contain Noun as a part of speech, whereas a COPE may be created for BareNoun InflectedNoun, or even BantuNoun. COPEs were conceived to separate theory specific knowledge (cf. the constructs of Minimalism or HPSG) from more broadly accepted knowledge. This is the basic idea behind a COPE. The case where knowledge needs expanding is relatively easy to deal with since knowledge is added monotonically without contradiction. A COPE is created and the appropriate concept (e.g., a new type of noun) is asserted as an extension of GOLD. As another example, consider the various linguistic properties: tense, case, aspect, person, number, etc. Their cross-linguistic relevance is mostly beyond dispute. This is why CaseFeature, TenseFeature, etc. are included as part of GOLD’s core inventory of concepts. However, the individual features for particular linguistic descriptions may differ significantly. This suggests, for instance, that past tense in, for example, German is different from the past tense of, say, Igbo. But in general, we know that tense is a general linguistic feature, represented as TenseFeature, and is applicable cross-linguistically. One way to capture individual tenses of particular languages is to extend TenseFeature with language specific classes (not instances of classes) such as GermanTenseFeature and IgboTenseFeature. At the very local level, specific tenses are then instances of those classes.

\[
\text{GermanTenseFeature(DEU\_PAST\_TENSE)} \quad (1.16)
\]

\[
\text{IgboTenseFeature(ibo\_PAST\_TENSE)} \quad (1.17)
\]

Thus, the concept TenseFeature is defined directly in GOLD. But language or description specific concepts such as GermanTenseFeature and IgboTenseFeature would be defined in corresponding COPEs. More detailed assertions can be made about specific classes in COPEs, such as “infinitives in German are not inflected for tense”.

The second case where additional knowledge contradicts the upper ontology resembles the problem encountered with canonical typology. This case is less straightforward, because a knowledge base should be free of contradiction. This principle applies to all types of knowledge, even to knowledge of the linguistic sort. As noted in Section 1.3, this is particularly true when a reasoner (e.g., a theorem prover) is to be used. But a reasoner need not be employed using every component of the knowledge base. If there is a canon stating that all lexical items of every part of speech have access to all feature types (see Corbett, this volume), then that portion of the knowledge base need not be factored in when a particular linguistic description is being considered. Of course this example represents knowledge which is simply a narrowing of assertions that are already present. But consider knowledge that is directly contradicted, such as the canon stating that “a lexical item belongs to just one part of speech” (also from Corbett, this volume).
In the case of mixed categories such as gerunds and participles, as Corbett points out, the canonical assertion is clearly violated.

One way of measuring deviation from the canon (“distance” in linguistic space) is to determine how many assertions are violated by the actual data description. But violations need to be of a particular sort, namely violations of canonical assertions, and not general assertions, such as “linguistic signs have meaning components”. What is proposed, then, is a framework that includes a COPE for canonical knowledge. The result is a knowledge base with at least three separable components or levels:

1. Upper ontology (inviolable assertions)
2. COPEs (mid-level assertions)
   - Canonical COPE (likely violable assertions)
   - Theory-specific COPE
   - Area-specific COPE
3. Instance data (local assertions)

The first level is the upper ontology that contains only most general sort of knowledge that should not be violated—basically what GOLD is intended to be. The second level is the COPE. COPEs can be of any type, not just for canonical assertions. A theory- or area-specific COPE (e.g., HPSG or Grassfields respectively) may only be applicable to a limited set of data. Third is the level of instance data, or the local assertions that apply only to specific language varieties or to individual descriptions of language varieties. Instance data instantiates classes from any COPE except for the canonical COPE. This accords with the following from Nikolaeva (this volume): “[l]ike comparative concepts, canons are linguists’ constructs and arguably not part of the speakers’ grammar.” That is, concepts from a canonical COPE will not be fully instantiated by real world data. Of course, all languages will exhibit canonical properties along particular dimensions but certainly not for all dimensions.

1.5 Encoding Descriptive Data

This section provides a discussion of how individual descriptions of language data can be transformed to a knowledge base structured according to GOLD, and perhaps according to a number of COPEs. First the problems associated with lexical data are discussed, followed by a discussion of interlinear glossed text. Finally, we give a brief overview of Linked Data, the technology behind storing such data in on the Web.
1.5.1 Lexical data

In order to model lexical data in GOLD, several new concepts need to be introduced. First is the lexeme itself. In GOLD the concept Lexeme is defined as a set of sign instances all having the same basic sense. Thus, if LEX123 were the entry for the English lexeme fish, members of LEX123 would be any instance of the sign FISH (i.e., all occurrences of fish, written or spoken). And, since the sign FISH belongs to the language variety ENGLISH, a new subclass of Lexeme can be derived, namely EnglishLexeme. In this way any specific assertions concerning the English lexeme can be encoded in a separate COPE, without affecting other types of lexemes. In summary, the following concepts are necessary to model the notion of a lexeme:

- Lexeme: the concept of lexeme (subsuming individual language lexemes)
- hasMember: the binary relation between a lexeme and a given sign instance
- EnglishLexeme: the concept of an English lexeme

The following assertions give a clear idea of how this knowledge is instantiated as part of a COPE for the English lexicon:

\[
\begin{align*}
\text{EnglishLexeme} & \subseteq \text{Lexeme} \\
\text{EnglishLexeme}(\text{LEX123}) & \\
\text{hasMember}(\text{LEX123}, FISH1) & \\
\text{orthoForm}(FISH1, \text{“fish”})
\end{align*}
\]

The first statement declares EnglishLexeme to be a subclass of Lexeme. The second statement declares LEX123 as an instance of EnglishLexeme. The next statement associates the sign instance FISH1 with the lexeme LEX123. Finally, the orthographic form for the sign instance FISH1 is given.

In terms of migrating lexical data to GOLD, the main issue concerns how to transform the dictionary form (a display-oriented data format) into explicitly structured knowledge. The task includes deciding how particular descriptive data will be instantiated according to the concepts in the ontology, i.e., according to GOLD or to a COPE. Consider a lexical entry in Table 1.1 for the language Potawatomi\(^3\) (ISO 696-3: pot):

First, we should distinguish the structuring aspects of the entry from the entry’s contents. An entry contains a headword and a definition as structuring devices, things that are particular to this type of linguistic data structure. The linguistic data content packaged in the entry consists of actual instances of linguistic signs. With respect to a lexical entry, an analogy from computer science would be the

---

\(^3\) Data was provided by Laura Buszard-Welcher, posted on-line at http://emeld.org/school/classroom/stylesheet/potawatomi1.html
1.5 Encoding Descriptive Data

array that contains a company’s employee IDs, versus the IDs themselves. The structural components of the lexical entry correspond to the array, while the linguistic signs correspond to the employee IDs. The IDs are, thus, the content, while the array is the structuring device.

How to separate data structuring devices from data content is sometimes not so clear. But by comparing different kinds of linguistic data types, as shown in Table 1.2, we get a sense of what is shared across them, namely linguistic signs. That is, if the same language data were found in another type of data presentation format, say a morphological paradigm, then whatever is shared among the two data types would likely be the language data, and not relevant to a particular presentation format. Particular linguistic forms are the most important elements of these data types. On the other hand, structural components do not matter for language itself, rather only for its description and presentation. Put yet another way, we see the orthographic representations for several linguistic signs, but these signs are playing particular roles within the given data type. For instance, we have bgan as the headword, but also the various inflected forms, bganen, nbeganem, etc. as exemplars. Finally, we have English linguistic signs as the definition. Notice that in a monolingual Potawatomi dictionary, a form such as bgan could be used just as easily in the definition as it could be in the headword.

<table>
<thead>
<tr>
<th>linguistic data type</th>
<th>data type component</th>
<th>content</th>
</tr>
</thead>
<tbody>
<tr>
<td>IGT</td>
<td>gloss, translation</td>
<td>linguistic sign, linguistic feature</td>
</tr>
<tr>
<td>lexical entry</td>
<td>headword, definition</td>
<td>linguistic sign, syntactic category</td>
</tr>
<tr>
<td>paradigm</td>
<td>entry</td>
<td>linguistic sign, linguistic feature</td>
</tr>
</tbody>
</table>

TABLE 1.2 Anatomy of linguistic data types

That leaves the various linguistic properties, expressed in Figure 1.1 as pl, pos, etc. These are of course specialised terms used to describe language data, part
of a controlled vocabulary for general linguistic description. Unlike the notions of headword and definition, the specialised terminology concern a specific linguistic analysis, much like a transcription or syllabification (not actually shown in the entry). We encode such expression of linguistic analysis as properties of the linguistic sign.

The first set of statements assert the types for the various linguistic signs: \textit{SyntacticWord} is a subclass of \textit{LinguisticSign}; the sign is associated with a particular orthographic form:

$$\text{SyntacticWord}(\text{POT\_BGAN\_1})$$ (1.22)
$$\text{orthoFrom}(\text{POT\_BGAN\_1, “bgan”})$$ (1.23)
$$\text{SyntacticWord}(\text{POT\_BGANEN\_1})$$ (1.24)
$$\text{orthoFrom}(\text{POT\_BGANEN\_1, “bganen”})$$ (1.25)

Next, the lexeme and its members are asserted. Note that the members of the lexeme are given as syntactic words, but this could be any number of sign subtypes such as \textit{Stem}, \textit{InflectedForm}, etc.

$$\text{PotawatomiLexeme}(\text{POT\_BGAN\_LEXEME})$$ (1.26)
$$\text{hasMember}(\text{POT\_BGAN\_LEXEME, POT\_BGAN\_1})$$ (1.27)
$$\text{hasMember}(\text{POT\_BGAN\_LEXEME, POT\_BGANEN\_1})$$ (1.28)

According to the entry, the various signs have linguistic properties. This is expressed in the following statements:

$$\text{hasFeature}(\text{POT\_BGAN\_1, POT\_SG})$$ (1.29)
$$\text{hasFeature}(\text{POT\_BGANEN\_1, POT\_PL})$$ (1.30)

1.5.2 \textit{Interlinear glossed text data}

Turning to another data type, interlinear glossed text (IGT), we observe a similar situation, where this data structure is a collection of linguistic signs packaged into a three-line display format. Consider this very simple example:

(4) \textit{los} perro\textit{-s}
\textit{DT.PL} \textit{dog} \textit{PL}
‘the dogs’

On the surface, we observe in (4) only linguistic signs, delimiters, and specialised terminology, but arranged in a particular way. The required structural components for simple IGT are in fact (1) various delimiters indicated by the hyphen and period, (2) the glosses (abbreviations for specialised terms) and (3) a translation.
In the example, the sign \textit{the dogs} plays the translation role, while $DT.PL$, $dog$, and $PL$ are glosses. The representation \textit{los} corresponds to an individual instance of \texttt{LinguisticSign}, actually an instance of \texttt{SyntacticWord} call it LOS1. The statements below provide further detail:

\begin{align*}
\text{SyntacticWord}(\text{LOS1}) & \quad (1.31) \\
\text{Stem}(\text{PERRO1}) & \quad (1.32) \\
\text{InflectionalUnit}(\text{PLURAL_S1}) & \quad (1.33)
\end{align*}

Any level of morphosyntactic detail is possible, but the entire linguistic sign \textit{los perros} should be declared:

\begin{align*}
\text{Phrase}(\text{PHRASE123}) & \quad (1.34) \\
\text{orthoFrom}(\text{PHRASE123}, \text{“los perros”}) & \quad (1.35)
\end{align*}

In this way a translation can be given and related to the phrase:

\begin{align*}
\text{Phrase}(\text{PHRASE456}) & \quad (1.36) \\
\text{orthoFrom}(\text{PHRASE456}, \text{“the dogs”}) & \quad (1.37)
\end{align*}

\begin{align*}
\text{translationOf}(\text{PHRASE123}, \text{PHRASE456}) & \quad (1.38)
\end{align*}

Finally, we specify the linguistic features that are associated with each sign. For instance, assuming that part of speech and number are features, we have:

\begin{align*}
\text{hasFeature}(\text{LOS1}, \text{PluralNumber}) & \quad (1.39) \\
\text{hasFeature}(\text{LOS1}, \text{Determiner}) & \quad (1.40)
\end{align*}

As the number of statements demonstrate, the amount of information packed into a single simple example of IGT is large, perhaps larger than expected. This is because so much is assumed in the compact display-oriented data structure. The above statements are given in a Description Logic formalism which is rather compact, although in an actual Web implementation, the compact nature of the representation is lost. Such is necessary if the data and meaning of annotation are to be made explicit and linked together as a coherent whole. The approach, known as Linked Data, is explained in the following section.
1.5.3 Linked Data

The purpose of encoding descriptive data is to make every assumption explicit especially regarding the meaning of the annotations. Two important criteria are required for an implementation: (1) uniqueness both of annotation elements and data content and (2) the ability to place such information on the Web and link among individual data. Uniqueness refers to the identifiability an individual datum at any level of granularity. That is, a single phonetic feature instance should be as uniquely identifiable as an entire text. We want the Web to be the medium of data storage and the ability to link among that data.

Uniqueness is achieved by using Uniform Resource Identifiers (URIs) for all data (Berners-Lee et al., 1998). A URI is a reference scheme that can be used to refer to anything whatsoever, from documents on the Web to actual physical entities (e.g., my horse) or even abstractions (e.g., world peace). As a naming scheme, URIs use the typical http protocol of the Web. For instance, the following URIs refer to linguistic constructs within the GOLD ontology:

- http://purl.org/linguistics/gold/PluralNumber
- http://purl.org/linguistics/gold/Lexeme
- http://purl.org/linguistics/gold/LinguisticFeature

Note that a URI does not need to actually be located “on the Web”. The following URIs are perfectly valid identifiers (for the obvious referents):

- http://solar-system.org/Mars
- http://people.com/JohnSmith
- http://ideas.co.uk/WorldPeace

URIs are only the first requirement for the implementation. The second is a way to link the data. For this we use the Resource Description Framework (RDF) (Lassila and Swick, 1999) which is basically a graph model with a serialisation (physical representation) amenable to the structure of the Web. Within an RDF graph, we require that graph nodes and arcs all have URIs. Thus, every node and arc is identifiable and definable. The ontological framework is built atop the RDF/URI system and statements are made using URIs as predicate and argument. The basic element of an RDF graph is the triple which is of the form subject-predicate-object. The following statement can be represented as an RDF triple:

\[
\text{hasFeature}(\text{LOS1}, \text{Determiner})
\]

Each concept, role and individual has a URI and occupies a node (or arc) in the RDF graph. The entire enterprise of placing data on the Web using URIs and RDF is known as the Linked Data approach (Berners-Lee, 2006).
1.6 Summary and Discussion

Our main goal in this chapter was to contribute to an understanding of how canonical typology might be implemented using an ontology-driven approach. As the primary framework for discussing the various issues, we used the General Ontology for Linguistic Description. To present various example assertions, we used Description Logic formalism throughout. The main argument was that an ontology-driven approach to typology resembles the canonical enterprise in several ways. In particular, we showed that the metaphor of the canonical space has been formalised in other (non-linguistic) ontological frameworks. We argued that the idea of a Community of Practice Extension, a component already established within the ontological framework, can be used as a device to compare individual language descriptions to pre-defined canonical knowledge.

Also for the purpose of providing concrete examples, we showed how two types of descriptive data (lexical entries and interlinear glossed text) could be encoded in an ontological framework. We discussed how the amount of information packed into these display oriented structures is actually more than meets the eye and leads to a proliferation, but a high degree of explicitness, of information. Finally, we discussed the technology used to implement rich information, that of Linked Data which is based on Uniform Resource Identifiers (URIs) and the Resource Description Framework (RDF).

A key open question relates to search. Starting with an inventory of descriptive data, how can those data be searched to discover how they compare to the canon. For instance, examples of such searches (given in prose) might be to show:

- the most canonical example of accusative case given several descriptions
- the least canonical example of negation given several descriptions
- the most canonical example of finiteness according to the dimension of tense marking
- examples of how gender is morphologically realised, from the most to the least canonical

The formalisation and execution of such queries present significant technical challenges, the main issue being how non-discrete operations are to be handled. For instance, the ontological machinery discussed here can only be used within particular discrete reasoning systems. Thus, scalar operations are not currently factored into the framework. That is, there is no way to state that the tense system in Finnish is, say, 7 units away from the canon for tense systems, while that of English is 8 units away. It is not even clear what such a unit would be. Again, the issue relates to the notion of a canonical “space”. We suggested that a discrete reasoner could be used to measure the number of contradictions of canonical assertions (contained in a canonical COPE), and perhaps this number could be one form of calculating canonical distance. For the solution, the next area to explore is that of probabilistic knowledge representation and reasoning.
Within such systems, probabilities can be assigned to each assertion. Contradic-
tions are essentially allowed: they just have a very low probability. Though likely
inadequate for more complex queries, the present framework provides a starting
point for further research into how canonical typology can be implemented and
how linguists can start to tackle the ever the problems created by the ever growing
amount of descriptive data in digital form.
References


References