Linguistic Issues in Language Technology – *LiLT*
Volume 1, Issue 1 June 2008

Constructing Integrated Corpus and Lexicon Models for Multi-Layer Annotation in OWL DL

Aljoscha Burchardt
Sebastian Padó
Dennis Spohr
Anette Frank
Ulrich Heid

Published by CSLI Publications

Constructing Integrated Corpus and Lexicon Models for Multi-Layer Annotation in OWL DL

Aljoscha Burchardt, *Technische Universität Darmstadt,* Sebastian Padó, *Stanford University,* Dennis Spohr, *Universität Stuttgart,* Anette Frank, *Universität Heidelberg,* Ulrich Heid, *Universität Stuttgart*

Abstract. We present a general approach to formally modelling corpora with multi-layered annotation in a typed logical representation language, OWL DL. By defining abstractions over the corpus data, we can generalise from a large set of individual corpus annotations, thereby inducing a *lexicon model.* The resulting combined corpus and lexicon model can be interpreted as a graph structure that offers *flexible querying functionality* beyond current XML-based query languages. Its powerful methods for *characterising and checking consistency* can be used for *incremental model refinement.* In addition, the formalisation in a graph-based structure offers the means of defining flexible *lexicon views* over the corpus data. These views can be tailored for linguistic inspection or to define clean interfaces with other linguistic resources. We illustrate our approach by applying it to the syntactically and semantically annotated SALSA/TIGER corpus, a collection of German newspaper text.

LiLT Volume 1, Issue 1, June 2008.
Constructing Integrated Corpus and Lexicon Models for Multi-Layer Annotation in OWL DL.

1 Introduction

Over the last two decades, considerable effort has gone into the creation of corpora with linguistic annotation. Corpora of written text have been annotated with a wide range of linguistic levels ranging from "shallow" morphological tags to "deep" semantic and rhetorical information; speech corpora have been enriched with phonetic transcriptions as well as suprasegmental analyses.

An increasing number of corpora contains analyses of more than one linguistic level. This group is commonly called corpora with *multiple layers of annotation* or *multidimensional markup* (Ahn et al., 2006). Well-known examples for English are the Penn Treebank (Marcus et al., 1993), which originally combined morphological and syntactic annotation and has since been extended with semantic roles (Palmer et al., 2005) and discourse relations (Miltsakaki et al., 2005), or the OntoNotes project (Hovy et al., 2006). Increasingly, such resources become also available for other languages. Examples include the Prague Dependency Treebank (Hajičová, 1998) for Czech and the SALSA corpus (Burchardt et al., 2006) and Potsdam Commentary Corpus (Stede, 2004) for German.

The creation of multi-level corpora makes efficient use of existing resources, since annotation of "deeper" levels typically presupposes at least a certain amount of more "shallow" analysis. In the Penn Treebank, for example, semantic and discourse annotation both refer back to constituency information. This allows the addition of specific types of annotation even with limited resources. More importantly, multi-layer corpora provide a richer linguistic structure than corpora with single annotation: They offer a wealth of data on the *interface and mapping phenomena* between different linguistic levels.

These data provide an excellent opportunity for *descriptive linguistic modelling* of these interfaces, that is, to create linguistic models and theories informed by the observed corpus patterns. Even though traditionally a contentious issue, the value of corpus evidence and quantitative assessments for descriptive modelling is now widely recognised, in particular in linguistic fields where analyses are highly context-dependent, such as semantics and lexicography (Fillmore, 1985, Kilgarriff, 1997a, Hanks, 2000). However, it appears that the potential of multi-layer corpora in this respect has not yet been fully realised.

The use of multi-layer corpora for Natural Language Processing purposes is not without its problems, either. Few studies have managed to integrate information from multi-layer corpora into large symbolic systems and processing architectures. There are a number of successful

statistical models, for example for the interface between syntax and semantic roles (Gildea and Jurafsky, 2002, Carreras and Màrquez, 2005). However, linguistic generalisations and interpretations remain implicit in the models and are hard to recover in symbolic form. Also, statistical models tend to have difficulty in distinguishing rare valid patterns from noise. Since there is evidence that ignoring infrequent events can harm performance (Daelemans et al., 1999), they could conceivably profit from cleaner and more linguistically interpreted data.

We attribute this state of affairs to three main *methodological challenges in using multi-layer corpora* that represent obstacles on the way to a better usage of these corpora:[1]

Querying multiple annotation layers. An immediate difficulty that users of multi-layer corpora face is getting efficient access to meaningful corpus patterns, which may combine specifications on multiple levels of analysis. Since it is unreasonable to assume that all such patterns can be pre-specified, this requires the ability to specify *arbitrary declarative* queries over all annotation levels. We find that current approaches either lack expressivity (in particular, with respect to intersective hierarchies and negation) or the ability to specify queries declaratively. Section 2 gives more details on this issue.

Representing hierarchical relations between annotation categories. Linguistic annotation schemes vary widely in the *granularity* with which they describe phenomena. Even within tagsets, categories are often hierarchically related, since there hardly ever is a single "right" level of granularity. For example, the Penn Treebank distinguishes between different types of adverbial phrases (`ADVP-TMP`, `ADVP-MNR` and others), which are clearly subtypes of the general type `ADVP`. Across layers, differences in the granularity mean that descriptions of interface phenomena appear more heterogeneous than they actually are. For example, when a semantic argument can be realised by individual words (such as interrogative and demonstrative pronouns) or by constituents (NPs, PPs), the extraction of syntax-semantics mappings from corpora can result in idiosyncratic patterns (Frank, 2004, Babko-Malaya et al., 2006). A similar problem arises in the integration of data from multi-layer corpora in ontological and deep grammar resources, whose granularity is typically fixed (Frank et al., 2007). Even in purely statistical applications, the choice of an appropriate level of granularity – which is not necessarily the one offered by the corpus – can make a significant difference in performance (Klein and Manning, 2003).

[1] All three challenges apply also to single-layer corpora; however, they are generally more serious in the case of multi-layer corpora.

Dynamically extracting and verifying regularities. One of the central goals of descriptive linguistic modelling is the extraction of valid generalisations. However, generalisations are rarely identified correctly at the outset. As a rule, they are the result of an incremental refinement process that takes into account corpus examples as well as external knowledge. This process could arguably benefit from a *dynamic* corpus model that (i) enables the efficient identification of "candidate generalisations" and (ii) can *enrich* the corpus with such candidate generalisations in order to identify counterexamples, inspect them and classify them as either genuine problems or subregularities.

An important application of this process is consistency control. Due to the complex interaction of linguistic levels, manually defined annotation schemes are often insufficient to ensure consistency. For example, the annotation of semantic roles requires a large number of categories that are frequently lexically specific and are often subject to inter-category relations such as obligatoriness or mutual exclusion. Therefore, it is very difficult, if not impossible, to specify all properties of each annotation category in an annotation scheme. A number of studies have shown that consistency can be improved considerably by extracting generalisations from corpora and treating them as well-formedness constraints (Dickinson, 2005, Hepple and van Genabith, 2000). Ideally, a dynamic corpus model should support this process without the need for additional tools through *corpus-driven enrichment of the annotation scheme.*

In this article, we present a general approach to formally modelling corpora with multi-layered annotation in a typed logical representation language, OWL DL. The graph structure that is defined by the OWL DL model has the potential of addressing the three challenges outlined above by virtue of being interpretable as an *integrated corpus and lexicon model.* It accommodates information about instances, external regularities, and data-driven generalisations on an equal footing and thus serves as a flexible basis for linguistic research and for natural language processing. More specifically, it offers *flexible querying functionality* beyond current XML-based query languages, can concurrently represent different degrees of granularities for *hierarchically related annotation categories*, and has powerful methods for *characterising and checking consistency* that can be used for *incremental model refinement.* Furthermore, it offers the means of defining flexible *lexicon views* over the corpus data. These views can be tailored for linguistic inspection or to define clean interfaces with other linguistic resources.

Note that our approach is orthogonal to what is generally discussed

under the heading of standardisation (Romary and Ide, 2004), since we assess representations primarily with respect to their suitability for *linguistic interpretation*. Thus, our modelling framework is applicable to multi-layered corpora irrespective of the particular corpus encoding scheme chosen.

Structure of the article. We first discuss current representation schemes, in particular XML-based storage technologies (Section 2). Next, Section 3 describes the representation formalism, OWL DL, and the design of integrated corpus and lexicon models in this formalism. The rest of the paper demonstrates the benefits of such models in the form of a case study on the SALSA/TIGER corpus, a multi-layer corpus with syntactic and role-semantic annotation. We first provide information on this corpus and describe the creation of the formal model (Section 4). Sections 5 through 7 demonstrate how the model addresses the three challenges introduced above. After a short discussion of related approaches in Section 8, we conclude (Section 9).

2 Current Representation Schemes

XML-based technologies. The current state-of-the-art representation for multi-layer corpora is stand-off annotation (Thompson and McKelvie, 1997), where each linguistic level is encoded separately. The levels are synchronised by virtue of using common identifiers to refer to linguistic entities. There are several representational models with corresponding tools to create and query corpora with linguistic annotation; examples are ATLAS (Laprun et al., 2002), which is based on and extends the Annotation Graph model (Bird and Liberman, 2001), and the NITE XML Toolkit[2] (NXT; Carletta et al., 2003).

In a survey carried out in 2004, Lai and Bird compared and analysed the most popular query tools on a number of queries (cf. the first challenge mentioned above). They found that some of such tools could not deal with intersecting hierarchies, i.e. tree-shaped analyses on multiple linguistic levels, which are ubiquitous in corpora with multi-layer annotation. Where such queries were possible (as in NXT), there were restrictions in terms of handling quantification and negation. Generally, they found that the relation of the query languages with database queries was not well understood, which raised concerns as to the scalability of these query tools to large datasets.

An alternative possibility is to query corpora with general-purpose

[2] `http://www.ltg.ed.ac.uk/NITE/`

XML tools, e.g., XSLT[3], XPath[4] or XQuery[5]. However, these tools are full-fledged programming frameworks (cf. Kepser, 2004). Their high expressive power means that they lose the advantages of linguistically oriented representation and query tools, such as relatively intuitive manipulation and task-driven representation. While recent proposals use these tools as an infrastructure to enable the representation and query of massively annotated corpora (e.g., Eckart and Teich, 2007), large-scale tests are still pending.

The second challenge, handling granularity, can be met in XML corpora to a certain extent. For example, it is possible to obtain coarser-grained representations procedurally by collapsing corpus annotation categories and encode these coarser-grained categories as additional attributes. However, this representation does not scale well to more than two granularity levels, and arguably does not provide a sufficiently general solution. These problems are also recognised in the XML community (Renear et al., 2002).

The final challenge, dynamic integration of regularities, is also difficult. XML storage is typically static. While regularities can be formulated as queries over the corpus, there is limited support for storing regularities together with the corpus. Thus, declarative consistency checking is limited to generic mechanisms such as DTDs or XML Schemas.

Relational databases. A relational database system (RDBMS) is potentially a very attractive option for storing data. Relational databases provide a well-researched, robust framework that directly incorporates solutions for many issues, such as consistency checking and efficient querying. However, the direct encoding of linguistic data in a relational database can be counterintuitive. For example, relational databases require the decomposition of hierarchical structures into non-hierarchical chunks. Therefore, trees must be encoded as a set of individual edges with shared nodes. Also, the encoding must conform to database-specific design criteria to reap the optimal efficiency benefits.

For these reasons, recent work in the OntoNotes project, which has successfully harnessed the power of a relational database to store a corpus with multi-layer annotation (Pradhan et al., 2007), has introduced an "object layer" above the relational database. All communication with the database happens through this intermediate object layer which provides an appropriate level of abstraction for the manipulation of complex linguistic objects. However, the object layer is implemented

[3] `http://www.w3.org/TR/xslt`
[4] `http://www.w3.org/TR/xpath`
[5] `http://www.w3.org/TR/xquery`

as an object-oriented application programming interface (API) in the programming language Python, but does not provide a declarative representation of the stored annotations.

3 Graph-based Modelling of Multi-Level Corpora

Our proposal in this paper is to formalise linguistic annotation as an integrated, multi-layered graph model within a declarative logical framework. On a formal level, it is a representative of a recent development in the corpus world towards graph-based data models (Francopoulo et al., 2006, Ide and Suderman, 2007, Polguère, 2006, Romary and Ide, 2004, Spohr and Heid, 2006, Trippel, 2006). While these studies advocate this move mostly on formal grounds (e.g., the need for interoperability, or the availability of efficient graph algorithms), our focus is somewhat different: We see our graph-based model as a good match for the needs of declarative linguistic modelling and natural language processing that we have stated as challenges in Section 1. It maps out a "middle ground" that combines the user-friendliness of existing XML-based representations with the expressiveness and efficiency of database-based approaches.

3.1 OWL DL as a Representation Formalism

The Web Ontology Language (OWL[6]) is a strongly typed monotonic formalism that extends the RDF Schema language (RDFS). OWL offers three different sublanguages with increasing expressivity – Lite, DL, and Full. Of these, the Description Logic sublanguage OWL DL is widely used in the Semantic Web community. It is also the most interesting one for our purposes, as it combines the expressivity of OWL with the features and favourable computational properties of description logics. For example, staying within the scope of expressivity of OWL DL ensures computational decidability (Baader et al., 2003).

The language itself can be expressed in XML syntax. Therefore, it should not be conceived of as a competitor of XML, but rather as a metalanguage whose semantics has been well investigated and is well understood, and whose vocabulary can be interpreted by a large number of existing tools (see below).

The three central representational means which constitute an OWL DL model are termed *classes*, *properties* and *individuals*. While classes represent concepts and can be instantiated by individuals, properties can either be used to link individuals in their domain to individuals in their range ('*object properties*'), or specify certain datatype values for

[6] `http://www.w3.org/2004/OWL/`

individuals in their domain ('*datatype properties*'). In addition, OWL DL offers the possibility to add logical restrictions to each class definition. These axioms typically serve the purpose of restricting the values that a certain property may have for individuals belonging to a particular class, or of constraining the types of individuals that may be instances of a particular class. The latter is achieved by defining a class as *disjoint* from another class, meaning that there cannot be an individual which is, e.g., an instance of both class A and a class disjoint from A. The definition of a property, on the other hand, can be constrained to be functional, symmetric, transitive, or the inverse of another property.

Both classes and properties are structured hierarchically, which means that the properties of the superclass are inherited by its subclasses. Moreover, since the formalism is monotonic, the axioms which have been defined for the superclass must also hold for its subclasses. OWL DL further supports multiple inheritance, so that a class may inherit from more than one superclass and, analogously, an individual may be an instance of more than one class. In properties, domain and range are inherited. This implies that an object property may not have a datatype subproperty, and vice versa.

Using OWL DL has a number of practical benefits. A large number of tools are able to interpret the vocabulary of OWL and to provide reasoning and consistency checking services. Examples are the freely available reasoners FaCT++[7] and Pellet[8], or the commercial RacerPro[9]. In addition, the widely used ontology editor Protégé (Knublauch et al., 2004) provides a powerful user interface for editing and visualising OWL models. Finally, the Sesame framework (Broekstra et al., 2002) with its powerful query language SeRQL (*Sesame RDF Query Language*) is an architecture for storing and querying OWL and RDFS data and offers, among other things, a database backend. In this manner, the OWL DL framework can automatically profit from the favourable properties of data storage in a relational database.

The combination of these factors – from the formal properties of OWL DL to the available reasoning, storage and querying services – makes the use of OWL superior to modelling solutions involving the definition of new XML formats. Moreover, the status of OWL DL as a standard "meta-language" that is *neutral* with respect to particular annotation schemes and standards ensures that our modelling framework can be applied to already existing corpora with multi-layer annotation.

[7] `http://owl.man.ac.uk/factplusplus/`
[8] `http://pellet.owldl.com/`
[9] `http://www.racer-systems.com/`

3.2 OWL DL for Integrated Corpus and Lexicon Models

The structural properties of OWL DL discussed in the previous section can be exploited to model linguistic corpora with multiple annotation layers and generalisations over them in a way that addresses the three challenges laid out in Section 1. We start will begin by discussing the benefits of OWL for such an integrated corpus- and lexicon model at a more conceptual level, taking up the challenges defined in Section 1. A concrete instantiation of such a model is then presented in Sections 5 to 7.

The first property of OWL DL that is beneficial for linguistic data is *multiple inheritance*, i.e., the ability of instances to instantiate more than one class. This ability is helpful for the modelling of multiple annotation layers with potentially overlapping hierarchies. For example, the annotated entities in a corpus can be represented in the model as having – in addition to syntactic categories – an arbitrary number of additional properties. Examples are lexical information (e.g., the semantic and syntactic heads) or functional classes (such as surface subject and deep object). In this manner, the functionalities offered by the OWL framework addresses our second challenge.

Next, OWL DL models can be understood as *graphs* – net-like structures with a high degree of interaction between layers of linguistic description. The nodes of the graph are formed by the entities in the model. Since each entity instantiates one or more classes, it has a number of *properties* which can be interpreted as labelled edges connecting the nodes. For example, nodes representing words could instantiate a class `Word`, one of whose properties might be `hasPartOfSpeech`. The corresponding edge in the graph would link the word node to another node representing its part of speech. This graph structure also allows for an intuitive formulation of arbitrary queries (our first challenge) in terms of the specification of "interesting paths" in the graph structure. In fact, this approach is supported by SeRQL, an existing efficient and powerful query mechanism for OWL DL.

Finally, the bipartite structure of an OWL model is naturally mirrored in the structure typical for annotated corpora. The TBox houses the *class hierarchy* that describes the structure of the annotations and data-driven as well as externally motivated generalisations, while the ABox contains the actual *corpus instances*. The ability to add constraints to a class that have to be fulfilled not only by all instances of this class, but also by instances of all its subclasses, gives rise to a strong interaction between the class hierarchy and the corpus instances. This interaction allows us to address the third challenge, the dynamic

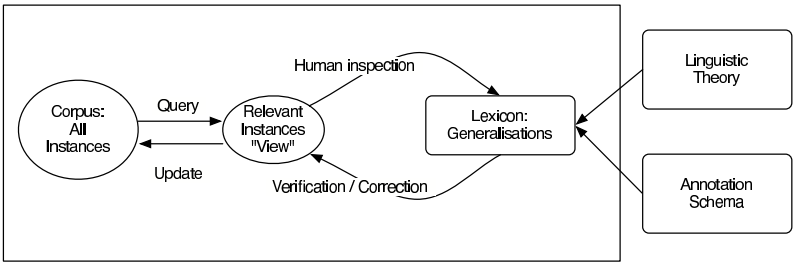

FIGURE 1 Incremental model refinement through interaction between instances and generalisations in the integrated corpus/lexicon model

extraction and verification of regularities.

Figure 1 illustrates the resulting *incremental model refinement*: The class hierarchy is initialised with the annotation schema plus possibly corpus-external linguistic considerations. Then, the corpus data can directly be checked for consistency with known generalisations. Due to the formal semantics that OWL DL provides, this further makes it possible to check constraints using general knowledge representation techniques. New generalisations can be acquired directly from sets of corpus annotations extracted with SeRQL, but also externally. This circle can be repeated as necessary.

3.3 Design Decisions for the Class Hierarchy

Figure 2 shows the technical details of the construction process of an integrated corpus and lexicon model. First, the class hierarchy is created from the annotation scheme and converted into an OWL DL "Model File" (TBox). Then, the annotated corpus instances are converted into OWL DL "Data Files" (ABox). After the consistency of the data files with this initial model has been checked, the combined corpus/lexicon model is stored in a Sesame RDFS database repository. This database serves as the basis for user interaction with the model. It supports both query-based extraction of information for linguistic enquiry (bottom right), and the model refinement cycle (shown with bold arrows).

In practice, the development of a class hierarchy that provides an optimal "backbone" for the organisation of the corpus instances from multi-layer corpora must take a number of linguistic and lexicographic considerations into account.

Implicit features in annotations. We stated above that the class hierarchy is induced primarily from the corpus annotation scheme(s). Linguistic annotation guidelines often concentrate on specifying the *lin-*

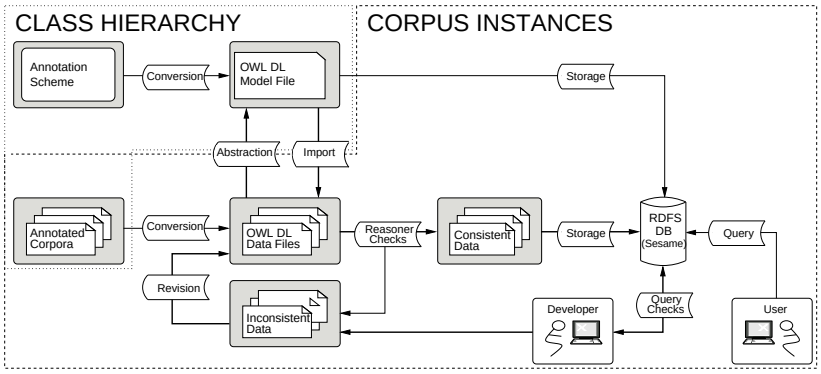

FIGURE 2 Workflow of the corpus/lexicon model creation process

guistic data categories to be annotated. However, a large part of the linguistically relevant information often remains implicit in the annotation scheme, and has to be inferred from particular configurations in the data. Even if such configurations can be inferred, it is often desirable from a practical point of view to make them more explicit, since the information can be vital in a number of respects, e.g. for defining clean generalisations in final lexicon resources (singling out "special cases"), for extracting information about special data categories, and for defining consistency constraints.

Lexicographic relevance and generalisations. A central requirement for any informative lexicon model is that it captures *knowledge about lexicographically relevant phenomena* in the corpus (see Atkins et al., 2003). Moreover, the model should support *abstraction* over specific annotation instances in order to derive further generalisations about these phenomena. As an example, consider valence patterns, which only arise clearly when information from a large number of corpus instances is combined.

Quantitative information. Closely connected to the previous issue is the ability to derive quantitative tendencies from corpus annotations, which is desirable both for statistical and for manual corpus analysis (Kilgarriff, 1997b). This task can be approached in two ways, both of which are supported by the combination of OWL DL and the Sesame framework (Broekstra et al., 2002). The *static* solution is to hard-code the frequencies of specific sets of phenomena in the lexicon. As a result, frequencies can be obtained directly as results of queries. However, the set of "countable" phenomena must be predefined in the class hierarchy

at corpus creation time, which limits the usability of the lexicon as a tool for active lexicographic and lexical-semantic research. In the alternative *dynamic* approach, the structure of the model and the query engine are designed so that quantitative tendencies can be derived at query time by grouping query results. This provides higher flexibility, but imposes higher demands both on the lexicon architecture, and on the skills of users.

4 Designing an Integrated Corpus and Lexicon Model for the SALSA/TIGER Corpus

This chapter describes the SALSA/TIGER corpus, a corpus with syntactic and role-semantic annotation, and the design of an integrated corpus and lexicon model for the corpus. This model will then be used as a concrete example to illustrate the benefits of an integrated corpus and lexicon model.

4.1 The SALSA/TIGER Corpus

The aim of the SALSA project (Burchardt et al., 2006) is to produce a large German corpus with role-semantic annotation. To this end, it extends the TIGER corpus, a German newspaper corpus with manual syntactic annotation (Brants et al., 2002), with a *role-semantic layer* in the Frame Semantics paradigm (Fillmore, 1985).

Frame Semantics is a semantic theory that describes the meaning of verbal, nominal, and adjectival predicates by their reference to *frames*, schematised situations. Each frame specifies the "participants and props" of the situation, which can be realised linguistically as semantic roles. The Berkeley FrameNet project (Fillmore et al., 2003) is working on a comprehensive description of the core lexicon of English in terms of frames and semantic roles. The semantic roles are defined at the frame level and are subdivided into "core" roles (participants and props pertaining to the particular situation) and "non-core" roles (adjunct-like information such as time or place). FrameNet currently covers over 800 frames and more than 10,000 lexical items. There is evidence that FrameNet is substantially (if imperfectly) language-independent (Boas, 2005). While there are clear cases of cross-lingual divergence, the majority of FrameNet frames carries over to German. Thus, SALSA re-uses English frames for German whenever possible.

While each instance of a predicate usually introduces a single frame, this is not always true. In cases of ambiguity, SALSA allows its annotators to annotate frame pairs as *underspecified* when the context does not support disambiguation. This is consistent with practice in word sense annotation (Kilgarriff and Rosenzweig, 2000). Another type of

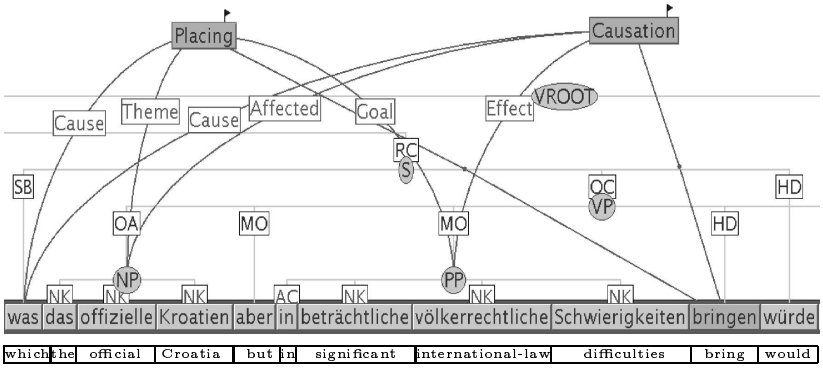

FIGURE 3 Multi-layer annotation of a German phrase with syntax and frame semantics *('which would bring the Croatian government into significant difficulties with international law')*

multi-frame annotation results from non-literal language use, a case of which is shown in Figure 3. The verb *bringen ('to bring')* is used metaphorically. Its "literal" reading is described by the frame PLACING which carries the inferences that a cause (*was/which*) is responsible for a theme (*Kroatien/Croatia*) ending up in a particular place, the goal (*in Schwierigkeiten/in difficulties*). The "understood" meaning is described by the frame CAUSATION, whose interpretation is that a cause imposes some effect on an affected party.

The SALSA corpus can serve as a representative instance of a multi-level corpus with intersecting hierarchies (arising from concurrent syntactic and role-semantic annotation) that faces the problems discussed in Section 1 (Heid et al., 2004). We therefore construct an integrated corpus and lexicon model in OWL DL for this corpus, using the complete SALSA release 1 as of September 2007 as our data. This release comprises approximately 20,000 annotated frame instances. Both the original corpus and our OWL DL model are available for download from `http://www.coli.uni-saarland.de/projects/salsa`.

4.2 Creating the Class Hierarchy

Following the procedure shown in Figure 2, the first step in model construction is the design of the class hierarchy (TBox) that describes valid annotations for the syntactic TIGER and semantic SALSA analyses. As discussed above in Section 3.3, we would like to capture generalisations not only over linguistic categories, but also over types of annotations (e.g., to characterise well-formedness). The class hierarchy we have created (partly shown in Figure 4) therefore consists of two parts. Each

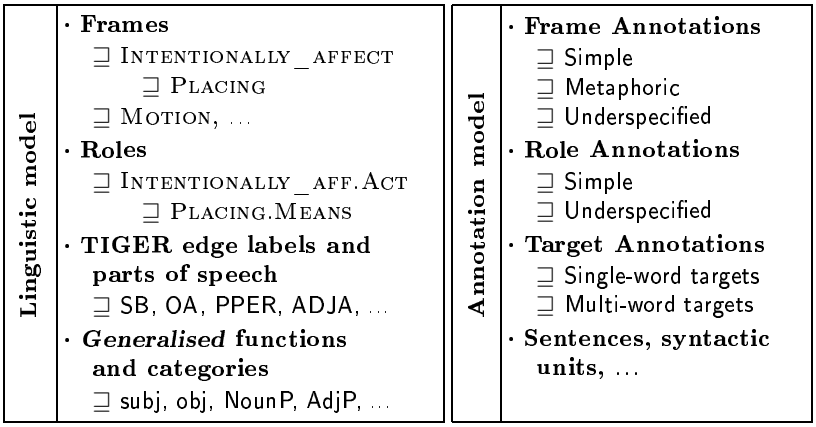

FIGURE 4 Schema of the OWL DL model's class hierarchy ("TBox")

corpus entity instantiates at least one class from each part.

The left-hand side of Figure 4 illustrates the *linguistic model*, in which frames and roles are modelled as classes, whose hierarchy reflects FrameNet's inheritance relations. Although this modelling decision seems to be straightforward, it is not the only possibility. Since FrameNet is a hierarchically structured resource with built-in inheritance relations, the user has the choice whether to model individual frames, such as SELF_MOTION or LEADERSHIP, either as instances of a general class FRAME in the ABox, or as part of the class hierarchy in the TBox (as we did). The first option results in a very lightweight class hierarchy which defines only the most basic concepts (frames and roles), while the complete FrameNet resource is modelled in the form of instances. The resulting model is simple, very flexible, and supports querying information about the frame types, such as their inherent semantic roles and relations. However, it gives the model little control over the actual annotation instances. Only if frames are modelled as a class hierarchy can the built-in reasoning mechanisms of OWL DL check the annotation and detect ill-formed annotations which, e.g., exhibit missing or superfluous roles. Queries that involve inheritance information, such as "find all instances of subframes of COMMERCE", are also considerably easier to formulate.

The right-hand side of Figure 4 shows a *hierarchy for annotation types*. It explicitly defines different classes of annotation phenomena and thus enables the search for entities that instantiate these types. For example, frame targets are marked as a multi-word target if their span contains at least two terminal nodes. Another annotation type that car-

ries specific properties is underspecification of frames and/or roles; only annotations of this type have the property `isUspWith`, which indicates the other participant in a pair of underspecified instances. Section 7 illustrates the use of annotation classes for consistency checking.

Both parts of the hierarchy demonstrate the use of the class hierarchy to encode hierarchical relations between annotation categories and annotation types. The addition of new levels of abstraction that are not present in the corpus annotation itself can be useful to derive concise accounts of phenomena. In our concept hierarchy, we have also added sets of (largely theory-neutral) grammatical functions and categories that subsume the fine-grained categories annotated by TIGER. These generalised categories support the extraction of generalised valence information from the lexicon, and their benefit will be illustrated in Section 6 below.

With respect to the representation of quantitative information, we opted for the more flexible solution of determining frequencies of phenomena dynamically through the arity of query answer sets.

4.3 Example Instance

To illustrate the discussion, Figure 5 shows a partial representation of the example from Figure 3 in the OWL DL model. The boxes represent instance nodes, with classes listed above the horizontal line, and datatype properties below.[10] The links between these instances stand for the OWL object properties defined for the instantiated classes.

The figure is centred around the metaphorical PLACING frame, the grey box in the middle of the figure, and highlights the model's graph-based structure with a high degree of interrelation between the lexicon entities. The grey PLACING frame instance is directly related to its roles (left, bottom), its lexical anchor (right), the sentence it is contained in (top), and a flag (top left, "Source") indicating metaphorical use. These entities are in turn all connected to further entities; for example, the sentence entity is connected to the literal frame (top middle) and the lexical anchor to the lemma and its sense (top right).

Multiple inheritance is indicated by instances carrying more than one class, such as the instance in the left centre which models the role-bearing constituent *das offizielle Kroatien*. It instantiates the classes `SyntacticUnit`, `NP`, `OA`, `NounP` and `obj`. Multi-class instances inherit the properties of each of these classes, so that e.g., the metaphoric frame annotation of the PLACING frame in the middle has both the properties defined for *frames* (`hasCoreRole`) and for *frame annotations*

[10] For the sake of simplicity, we excluded explicit 'is-a' links.

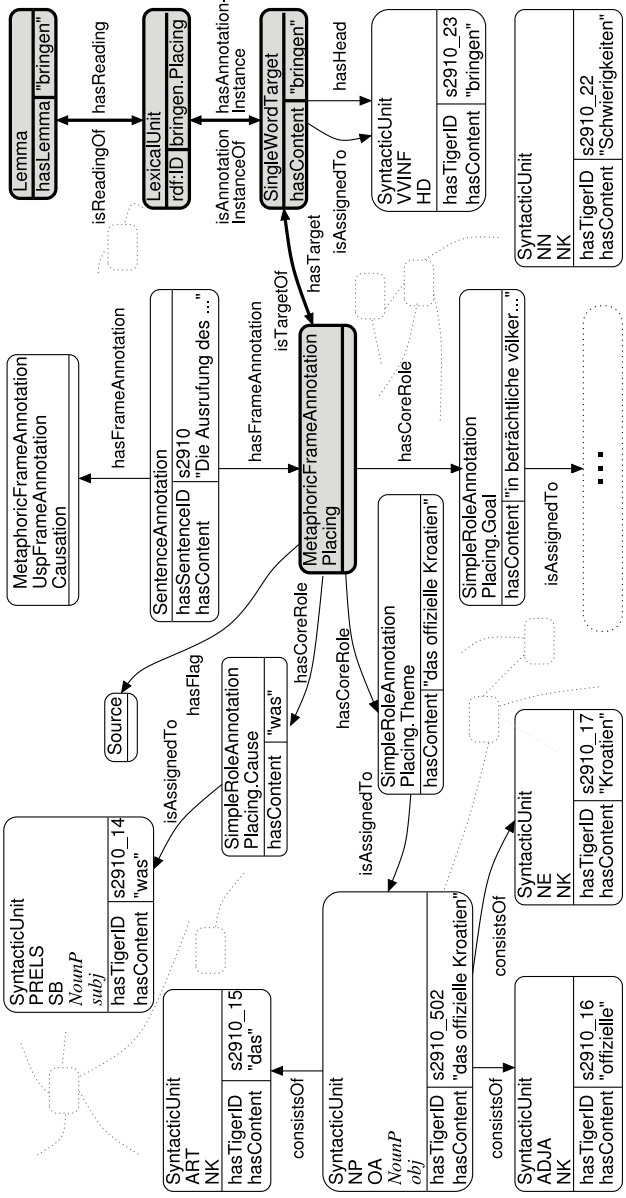

FIGURE 5 Partial lexicon representation of an annotated corpus sentence

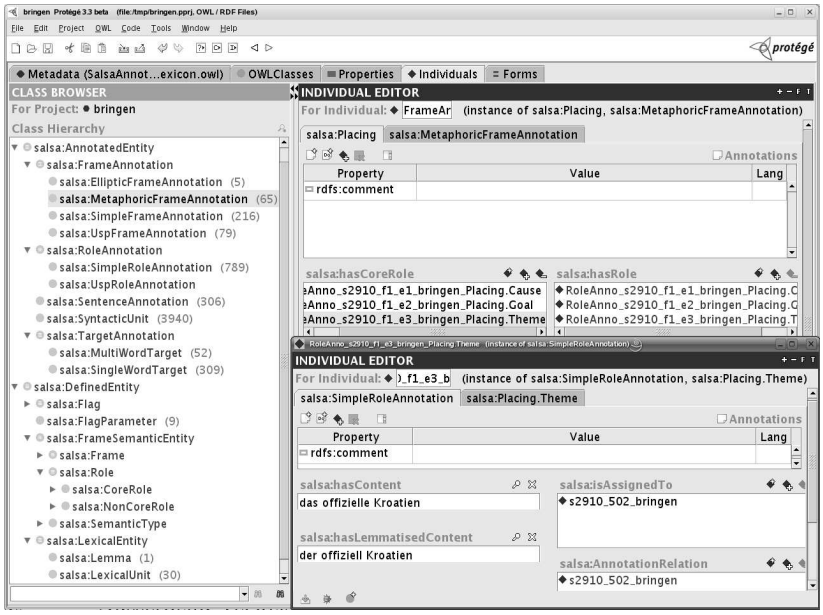

FIGURE 6 Viewing corpus data with the Protégé ontology editor

(`hasTarget`). The generalised syntactic categories discussed above are given in italics (e.g., *NounP*).

Viewing and editing the model. The lexicon model, as well as the instances, can be conveniently viewed and edited using the widely used ontology editor Protégé (Knublauch et al., 2004). Figure 6 shows a screenshot of Protégé. It displays the concept hierarchy on the left, along with instance counts for the individual classes. On the right, two nodes of the example instance from Section 4.3 are shown, namely the PLACING frame annotation instance (top), and its role annotation instance PLACING.THEME (bottom). The multiple inheritance as discussed in Section 4.3 is displayed as tabs (cf. *salsa:Placing* and *salsa:MetaphoricFrameAnnotation* in the top right window, and *salsa:SimpleRoleAnnotation* and *salsa:Placing.Theme* in the bottom right window). Each of these tabs shows only the properties that have been defined for the respective class, and thus, although the instance combines the properties of multiple classes, they are displayed separately according to their domain (cf. Section 3.1 above).

Figure 6 also illustrates how inferred information is displayed. Since the hierarchical organisation of both classes and properties contains

```
SELECT LEMMA
FROM {LEMMA} salsa:hasReading {} salsa:hasAnnotationInstance {}
             salsa:isTargetOf {} serql:directType {salsa:Placing}
```

FIGURE 7 SeRQL query that retrieves lemmas evoking the PLACING frame

the information that all "core roles" are "roles", all statements about the PLACING frame (top right) that involve the property `hasCoreRole` lead to the immediate inference of corresponding statements for its superproperty `hasRole`. Such inferred statements appear as grey items in the editor, as opposed to the asserted statements in black font.

5 Challenge 1: Querying

Recall from Section 4 that the OWL DL model is stored in the database that can be queried by the SeRQL language, which allows the user to search for arbitrary *subgraphs* in the corpus/lexicon model. The SeRQL language can use arbitrary combinations of properties and hierarchical information for the specification of these subgraphs. Figure 7 shows a simple example query which extracts all entities of type "lemma" that introduce the PLACING frame. The query can be understood as description of a path in the lexicon that begins at a lemma, follows a specified sequence of edges, and ends at a node that has the type PLACING, and through subtyping, the type "frame". The grey nodes in Figure 5 form such a path, making *bringen* a result for this query.

Syntactically, SeRQL is similar to other RDF query languages, such as RQL (*RDF Query Language*)[11] or RDQL (*RDF Data Query Language*)[12]. However, with its primary design goals being *"unification of best practices from query language [sic] and delivering a light-weight yet expressive query language for RDF that addresses practical concerns"* (Haase et al., 2004: p. 505), SeRQL has overcome some of the limitations of existing RDF query languages. It supports optional path expressions, datatype specifications, and reification (Haase et al., 2004)[13]. The Sesame architecture scales well (Guo et al., 2005) and is thus not subject to the efficiency concerns raised by Lai and Bird (2004).

As with all RDF query languages, the use of SeRQL does require familiarity with the class hierarchy to formulate effective queries; however, this is to be expected. One way to acquire the necessary knowledge is by examining the model in Protégé, or by browsing the items in the

[11] `http://athena.ics.forth.gr:9090/RDF/RQL/`

[12] `http://www.w3.org/Submission/2004/SUBM-RDQL-20040109/`

[13] This is also true for the the recently developed SPARQL (*Simple Protocol and RDF Query Language*), `http://www.w3.org/TR/rdf-sparql-query/`.

Lemma	*# instances*	*Rank*
legen (*'put'*)	38	1
bringen (*'bring'*)	35	2
nehmen (*'take'*)	13	3
plazieren (*'place'*)	4	4
ablegen (*'put down'*)	3	5
kippen (*'tilt over'*)	3	6
einführen (*'insert'*)	1	7
einlagern (*'stock'*)	1	8
einpflanzen (*'plant'*)	1	9

TABLE 1 Output after grouping the results of the query in Fig. 7

Type	*# instances*
Lemmas	523
Lemma-frame pairs (LUs)	1,176
Sentences	13,353
Syntactic units	223,302
Single-word targets	16,268
Multi-word targets	258
Frame annotations	16,526
Simple	14,700
Underspecified	995
Metaphoric	785
Elliptic	107
Role annotations	31,704
Simple	31,112
Underspecified	592

TABLE 2 Instance count based on the first SALSA release

HTML result page returned by Sesame. There is also always the possibility to define a set of "macros", i.e., expansion rules for frequent search patterns. Such rules can be integrated into a graphical user interface that further reduces the complexity of stating queries.

As discussed above, we can dynamically derive quantitative information by simple *grouping* of query results, i.e., by counting the frequencies of distinct results in the query's answer set. Table 1 shows the grouped results for the query from Figure 7, a frequency-ordered list of the German lemmas which can introduce the PLACING frame.

Using similar queries, we have computed the size of the overall SALSA/TIGER model. Table 2 shows that it contains a total of more

Role	*Specific (TIGER) POS/Category*	*Abstracted Category*
STATEMENT.SPEAKER	NN (noun)	NounP
STATEMENT.SPEAKER	NE (named entity)	NounP
STATEMENT.SPEAKER	PPER (personal pronoun)	NounP
STATEMENT.MESSAGE	S (sentence)	Sent
STATEMENT.MESSAGE	VROOT (tree root)	Sent

TABLE 3 Specific vs. abstract syntactic categories

than 304,000 instances of different types, instantiating 581 different frame classes and 1,494 role classes.

6 Challenge 2: Representing Hierarchical Relations

As we noted above, abstraction is a central challenge for the use of multi-layer corpora. When annotation categories are too fine-grained, patterns extracted from corpus data are often difficult to interpret. Additionally, frequencies for individual events tend to be low.

The *generalised classes* that we have defined in our model to abstract over the SALSA/TIGER corpus categories (cf. Section 4.2) can be used by queries to yield more compact patterns by aggregating annotated instances. As an example, Table 3 contrasts the results of a query that retrieves the syntactic realisation patterns for two semantic roles of the STATEMENT frame, with and without syntactic generalisation. We see that the number of unique categories is reduced from five to two. This trend holds in general: On the complete lexicon, the use of normalised categories reduces the number of realisation patterns from 5,065 to 2,289. Thus, the "generalised" patterns capture mapping generalisations which would have remained undetected otherwise. In addition, the general patterns are less tied to the particular set of parts of speech assumed by the TIGER treebank. This makes it easier, for example, to interface them with information from other resources, a task where granularity is a known issue (Frank, 2004, Babko-Malaya et al., 2006).

7 Challenge 3: Dynamically Extracting and Verifying Generalisations

Finally, we return to the main rationale of broad-coverage corpus annotation, namely the search for new insights and generalisations that should emerge from the annotated data. In this respect, our framework provides support for searching and extracting novel generalisations, defining continuous refinements of the model by introducing new

generalisations, and finally for controlling known generalisations that might be violated by annotation mistakes.

7.1 Lexicon Views

Virtually all lexicons, both computational or human-readable, follow a tree structure – they are lists of regularities structured according to a primary criterion (traditionally, the lemma). Entries may be further subdivided on the basis of secondary criteria (e.g., by word sense). While this representation scheme is easy to read, it is often the case for multi-level corpora that no single most important criterion can be identified. In the case of the SALSA/TIGER corpus, for example, frame-wise regularities may be just as interesting as statistics on the lemma or even role level, depending on the application.

The use of a graph-based corpus/lexicon model can avoid this problem, since the model itself, as a graph, does not need to commit to an individual hierarchical structure. By defining queries that target specific relations (such as between lemmas and their senses), we can then dynamically extract simpler subgraphs that are suitable, e.g., for linguistic analysis (cf. Section 5). We call the result sets of such queries *lexicon views*.

In the following, we contrast two lexicon views on the mapping between semantic roles on the one hand and syntactic categories and grammatical functions on the other. This mapping information can be used to validate linguistic theories on syntax-semantics mapping, for interfacing the annotation data with deep grammatical resources or general lexicons (Frank, 2004), or for learning automatic semantic role annotation systems (Carreras and Màrquez, 2005).

Table 4 shows a lexicon view on the mapping data that is structured by individual roles. For readability, we only show the data for the CAUSE_CHANGE_OF_SCALAR_POSITION reading of the lemma *senken ('to lower')*. Applied to our complete corpus, these views contain on average 8.5 role sets per lemma, and 5.6 role sets per frame. Note that this compact representation is due largely to our use of generalised syntactic categories such as NounP.

Table 5 demonstrates another possibility, namely the extraction of information about frame- or lemma-specific *role sets*, i.e., role realisation patterns for complete predicate-argument structures. The first row shows that the most frequent pattern seen for *senken* in this sense involved an AGENT role, realised as a (deep) subject noun phrase, and an ITEM role, filling the (deep) object noun phrase position. Again, the compact shape of this table is due to the syntactic generalisation: a query that contrasted abstracted (deep) syntactic information with

Role	*Category*	*Gramm. Function*	*Freq*
ITEM	NounP	obj	26
AGENT	NounP	subj	15
DIFFERENCE	PrepP	mod-um	6
CAUSE	NounP	subj	4
VALUE_2	PrepP	mod-auf	3
VALUE_2	PrepP	pobj-auf	2
VALUE_1	PrepP	mod-von	1

TABLE 4 Sample of role mapping patterns for *senken*/ CAUSE_CHANGE_OF_SCALAR_POSITION (CCSP)

Role set for senken / CCSP				*Freq*
AGENT		ITEM		11
subj		obj		
NounP		NounP		
	CAUSE	ITEM		4
	subj	obj		
	NounP	NounP		
		ITEM		4
		obj		
		NounP		
AGENT		ITEM	DIFFERENCE	2
subj		obj	mod-um	
NounP		NounP	PrepP	

TABLE 5 Sample of role set mapping patterns for *senken* / CCSP

corpus-specific labels and categories yielded 3,888 lemma-specific role sets for abstracted categories, but 7,790 for the specific classes. For frame-specific role sets, the figures are 3,125 and 6,875, respectively.

7.2 Incremental Model Refinement

Lexicon views can also serve another purpose, namely the *data-driven* search for linguistic generalisations which might not be obvious from a theoretical perspective. They also enable the quick inspection of data for counterexamples to regularities predicted by linguistic intuition.

In the case of semantic roles, such a plausible regularity is that within one lemma, each semantic role can occur as either (deep) subject or (deep) object, but not both. However, some of the role sets we extracted contained exactly such configurations. Further inspection revealed that these irregularities resulted from either noise introduced by errors in the automatic assignment of grammatical functions during the data conver-

sion process, or instances with syntactically non-local role assignments. Non-local role assignments result, e.g., from control/raising constructions or plausible inferences made by annotators (see Burchardt et al., 2006, for details), and can interfere considerably with the extraction of valence-based role set realisation patterns such as in Table 5.

This observation led us to refine our class hierarchy (cf. Section 4.2) by introducing a distinction between syntactically local and non-local role assignments into the annotation model. To this end, we defined an abstract class `hasPath` with the mutually disjoint subclasses `hasLocalPath` and `hasNonLocalPath`. Initially, all paths instantiated only the general `hasPath` property, corresponding to an "unresolved" path type. We now classified the path descriptions into local vs. non-local on the basis of path patterns by defining regular expression patterns for local path types on the basis of linguistic intuition. These patterns compiled into 216 distinct local path types, which we integrated into the model as instances of the subproperty `hasLocalPath`.

When we applied these patterns to the complete set of 30,772 path instances annotated in the entire corpus, we found that 25,779 (or 83%) of the paths were classified as local, corresponding to a high average of 119 path instances per path type. In stark contrast, the remaining 4,993 path instances (which are presumably non-local), are distributed over 1,311 distinct path types, with an average of 3.8 instances per type. That is, the set of non-matching paths is highly heterogeneous.

We then used the local path types to restrict the extraction of valence-based role set realisation patterns to rolesets with only local paths. We found that 93.8% of all roleset tokens were retained, while 6.2% were filtered out due to the presence of potentially non-local roles. While an exhaustive evaluation of the extracted rolesets is outside the scope of this paper, we evaluated two samples from both the set of "accepted" and "rejected" role sets. Of 80 randomly chosen local role sets, all were found to be correct. A sample of non-local rolesets showed a more varied picture: We found a mixture of infrequent syntactic constructions (such as the German *lassen* passive) and cases of truly non-local roles. An example of the latter case is shown in Figure 8, where the fact that "two-family homes" fills a semantic role of *buy* is a plausible, but defeasible, inference that should not be recorded as a generally valid role realisation pattern.

Our conclusion from this case study is that model refinement (in our case, by a simple characterisation of "local roles") can improve the quality of corpus-derived lexicon information. After one iteration of model refinement, all noisy instances have been removed from the extracted data. The resulting model is too "conservative" in the sense that

Besitzer von **Zweifamilienhäusern**, die vor 1987 *gekauft* haben
Owners of **two-family homes**, who before 1987 *bought* have

'Owners of two-family homes who have bought before 1987'

FIGURE 8 Example of an identified non-local role realisation: Role in boldface, predicate in italics (excerpt from TIGER s976).

some valid local paths are filtered out all as well. If desired, this can be changed by further iterations that extend the specification of local paths.

7.3 Consistency Control

An important motivation for the explicit representation of generalisations that we identified in Section 1 was the need for methods to formalise well-formedness, and thus to control consistency. Our OWL DL-based model offers two mechanisms for these tasks: axiom-based and query-based checking.

Axiom-based checking. Once some constraint has been determined to be universally applicable, it can be formulated in Description Logics as a well-formedness constraint on the annotation instances. For semantic role annotations, axioms can, e.g., define the admissible relations between a particular frame and its roles. This is illustrated in the DL statements below, which express that an instance of PLACING may *at most* have the roles GOAL, PATH, etc.

$$\mathsf{Placing} \sqsubseteq \exists.\mathsf{hasRole}\ (\mathsf{Placing.Goal} \sqcup \mathsf{Placing.Path} \sqcup \ldots)$$
$$\mathsf{Placing} \sqsubseteq \forall.\mathsf{hasRole}\ (\mathsf{Placing.Goal} \sqcup \mathsf{Placing.Path} \sqcup \ldots)$$

Relations between roles can be formalised in a similar way. An example is the *excludes* relation in FrameNet, which prohibits the co-occurrence of roles like CAUSE and AGENT of the PLACING frame. This can be expressed by the following statement.

$$\mathsf{Placing} \sqsubseteq \neg((\exists.\mathsf{hasRole}\ \mathsf{Placing.Cause}) \sqcap (\exists.\mathsf{hasRole}\ \mathsf{Placing.Agent}))$$

The restrictions are used in checking the consistency of the semantic annotation; violations of these constraints lead to inconsistencies that can be identified by a theorem prover (cf. the "Reasoner Checks" box in Figure 2).

While axiom-based checking is a very efficient and powerful method, its practical usefulness is currently limited. While experiments with the state-of-the-art theorem prover FaCT++[14] were indeed able to iden-

[14] `http://owl.man.ac.uk/factplusplus/`

tify some inconsistent role assignments, the current technology requires heavy user interaction to locate these instances. Given the prospect of advances in this area, especially concerning feedback given by the theorem prover, the ability to check for logical inconsistencies will be a definite asset of our approach in the future, minimising the effort necessary to guarantee the quality of manually annotated data.

Query-based checking. The current main method to check consistency in our corpus/lexicon model is query-based checking (cf. the "Query Checks" box at the right-hand side of Figure 2). The advantage of this procedure is that it can consider annotations in context and allows quick user interaction. It can be thought of as the flexible extraction of *"suspicious lexicon views"* (cf. Section 7.1), using SeRQL queries that can help to reduce or avoid manual effort. In this manner, the querying facility of our model provides a natural and general implementation of recent proposals for the detection of errors in linguistic annotation, which focus exactly on the extraction of "suspicious" patterns (Hepple and van Genabith, 2000, Dickinson, 2005).

Types of consistency checks

For the SALSA/TIGER data, we have identified three major classes of views which help identify annotation errors:

1. Controlling formal properties. This class controls *formal properties* of the annotation instances, as prescribed by the annotation guidelines. Such queries identify, e.g., inconsistencies in the assignment of the SUPPORTED role of support verb constructions – which ought to be assigned to the maximal syntactic constituent projected by the supported noun – or control a guideline to exclude reflexive pronouns from the span of the target verb. Applied to the SALSA/TIGER data, both queries yielded empty result sets – good evidence for the observance of formal annotation guidelines.

Further queries examine suspicious configurations of *annotation types*, such as target words evoking two or more frame annotations which are neither marked as underspecified nor tagged as a pair of (non-)literal metaphorical frame annotations. Here, we identified 8 cases of omitted annotation markup, namely 4 missing metaphor flags and 4 omitted underspecification links.

2. Consistency checking via FrameNet hierarchy. Sometimes, annotation errors are more subtle. For example, assume that we find an corpus instance of a predicate that introduces two underspecified frames f_1 and f_2 (cf. Section 4.1). If we also know that these frames are related by inheriting from a common frame f_0, it seems plausible that the frame elements of these two frames should be assigned consistently.

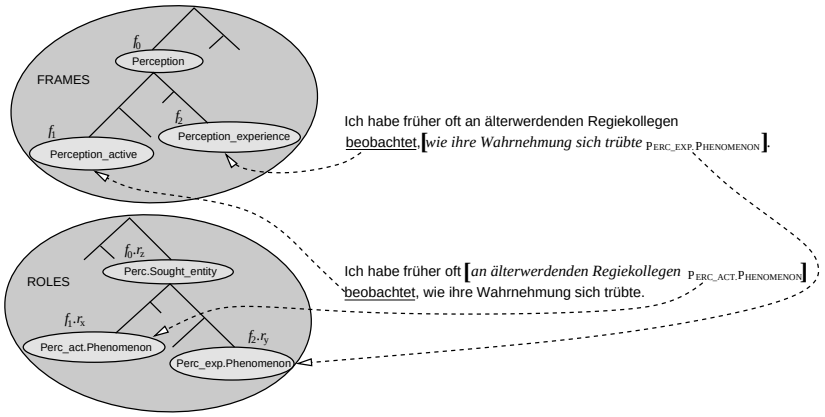

FIGURE 9 Detection of inconsistent role assignment. *('In the past, I have often noticed how the perception of ageing fellow directors began to blur.')*

More specifically, we require that all pairs of frame elements $f_1.r_1$ and $f_2.r_2$ which inherit from the same role $f_0.r_0$ are assigned to the same constituent.

We have formulated the negation of this rule as a SeRQL query, which returned 3 suspicious annotations out of the 38 instances matching the described configuration. All of these turned out to be role annotation errors; one is depicted in Figure 9. It involves the underspecified frames PERCEPTION_ACTIVE (introduced, e.g., by *watch*) and PERCEPTION_EXPERIENCE (introduced, e.g., by *see*). Both frames inherit from PERCEPTION. While the PERCEIVER_AGENTIVE / PERCEIVER_PASSIVE roles, which inherit from PERCEPTION.PERCEIVER, have been consistently annotated (not shown in the figure), the role PHENOMENON has been inconsistently assigned to *wie ihre Wahrnehmung sich trübte ('how their perception began to blur')* and *an älterwerdenden Regiekollegen ('ageing directors')*, respectively. Note that this query combines the full range of information encoded in the lexicon: frame and role annotations per se, the annotation type, and the complete FrameNet hierarchy with inheritance relations.

3. Extracting samples for manual outlier inspection. On the semantic level, we extracted annotation instances (in context) for annotation phenomena that are known to be error-prone. Examples include metaphorical vs. non-metaphorical readings, or frames that are involved in underspecification in certain sentences, but not in others. While the result sets that we obtain in this manner still require manual inspection, they are very compact, and their assessment is substan-

tially more efficient than direct inspection of individual annotation instances (Oliva, 2001, Dickinson, 2005). We see this result as concrete evidence of how the detection of inconsistencies can be enhanced by a declarative formalisation of the annotation scheme.

8 Related Work

The modelling of lexical resources has become a very active research area over the past years. In this section, we will relate our approach to the recent proposals and developments, both with respect to general lexicon modelling and the formalisation of frame-semantic information.

8.1 Lexicon Modelling

As mentioned in Section 3.2, all recent approaches to lexicon modelling are grounded in a graph-based view. One of the primary advocates of this direction is the *Lexical Systems* framework (Polguère, 2006), which aims at providing a highly general representation for arbitrary kinds of lexicons. As such, Lexical Systems should also be able to represent our data of interest, namely combinations of several corpus annotation layers. While this is technically true, we think that the formalism offers only insufficient structural means for the encoding of specific consistency constraints on the annotation layers and their interaction. In addition, Polguère emphasises the absence of hierarchical structuring of the data, saying that hierarchisation should be "projected on demand" (Polguère, 2006: p. 51). Other than Polguère (2006) and (Trippel, 2006, ch. 3), we start from linguistic models that contain a hierarchical organisation, which is also required in the context of semantic resources. In our application, the interaction between FrameNet's subsumption hierarchy and the semantic and syntactic annotation layers raises complex consistency issues (cf. Section 7.3 above), and clearly indicates the need for an expressive formalism to model hierarchical data.

Another recent proposal is the *Lexical Markup Framework* (LMF; Francopoulo et al., 2006), a currently evolving ISO standard for the modelling and the sharing of lexical resources. Its architecture provides for combination of different modules, each covering a specific field of linguistic description, which all relate to a common *core module*. We believe that our use of a typed formalism, which takes advantage of a strong logical foundation and the notions of inheritance and entailment (cf. Scheffczyk et al., 2006), is a crucial step beyond the representational means provided by LMF. Still, the Lexical Markup Framework, and with it the development of a standardised technical infrastructure that is to be expected in upcoming years, represents an important development towards interoperability between lexical resources. Considering

the descriptive procedures of LMF for describing our data format will be relevant in the future.

Finally, a close neighbour in terms of approaching the development of tools integrating lexicons and corpora is the *ATLAS* project (Architecture and Tools for Linguistic Analysis Systems; Laprun et al., 2002), whose objective is to represent and process annotation data in the framework of *Annotation Graphs* (Bird and Liberman, 2001). Although the primary focus of ATLAS is slightly different – it mainly deals with the annotation of corpus data in general, whereas we aim at deriving a lexicon model from corpus annotations –, the way the relevant issues are approached is very similar, e.g., in that they also provide for means to define different types of entities and assign to them properties with constrained sets of allowed values. ATLAS further combines annotations with a generic linguistic annotation ontology and a meta-annotation infrastructure (MAIA), similar to our formalisation of a *linguistic model* and several different *annotation types* on top of the actually annotated corpus instances (cf. Figure 4 in Section 4.2). To our knowledge, however, ATLAS currently supports only basic consistency constraints, and does not capture dependencies between different layers of annotation (cf. Laprun et al., 2002).

8.2 Modelling FrameNet

In the context of frame-semantic resources, three existing studies need to be mentioned, namely Narayanan et al. (2002), Baumgartner and Burchardt (2004), and Scheffczyk et al. (2006).

Baumgartner and Burchardt (2004) have modelled FrameNet in the framework of *Logic Programming* (see e.g., Grosof et al., 2003) in order to provide inference-based reasoning services. However, they have limited themselves to modelling the definitional part of the resource and did not include representations of annotated data.

Narayanan et al. (2002) have used DAML+OIL[15], a widely-used predecessor of OWL, in order to represent the first release of the FrameNet database. Their work was continued by Scheffczyk et al. (2006), who provided a formalisation of FrameNet in OWL and also included annotated corpus instances. However, their primary focus was on the ability to draw inferences from annotated text, or more generally, knowledge extraction. In contrast, we concentrate on constructing a rich linguistic representation to enable the linguistic exploration of the annotated data and their interrelations as well as the representation of appropriate generalisations.

[15] `http://www.daml.org/`

9 Conclusion

In this article, we have shown how a Description Logics-based, integrated corpus/lexicon model can be constructed directly from multi-layer linguistic corpus annotations. We have argued that such a model allows for the rich, explicit modelling of linguistic phenomena and for flexible and fine-grained definition of various degrees of abstractions over corpus annotations.

It can also overcome limitations of current XML-based search tools by supporting queries which are able to connect multiple levels of linguistic analysis in a declarative way. These queries can be used variously as an additional means of consistency control, to derive quantitative tendencies from the data, to extract lexicon views tailored to specific purposes, and finally as a general tool for linguistic research.

In conclusion, we note that this paper has mostly concentrated on consistency checking and revision of corpus data with completed annotation. An interesting direction for future annotation projects could be to check consistency already during the annotation process itself, and we think that our flexible, query-based method for consistency checking lends itself to this application. In an integrated annotation and querying environment, general consistency queries could be run at regular times to identify outliers for reannotation and resubmission. Both general and specific queries targeting specific annotation types could also be run immediately after the completion of a full annotation step. In conjunction with a feedback mechanism, the queries themselves can also be refined over time.

Acknowledgments

The research reported in this article was carried out in the context of the SALSA project at Saarland University, Germany, with which Aljoscha Burchardt, Sebastian Padó and Dennis Spohr were affiliated. Anette Frank was affiliated with DFKI Saarbrücken and Saarland University. The authors acknowledge the funding of Deutsche Forschungsgemeinschaft (grant Pi-154/9-2).

We would like to thank Andrea Kowalski and Katrin Erk for insightful discussions. We are also grateful to the audience at IJCNLP 2008 and to the editors and reviewers of LiLT, whose feedback helped to substantially improve the present article.

References

Ahn, David, Erik Tjong Kim Sang, and Graham Wilcock, eds. 2006. *Proceedings of the EACL Workshop on Multi-Dimensional Markup in Natural*

Language Processing, Trento, Italy.

Atkins, Sue, Charles J. Fillmore, and Christopher R. Johnson. 2003. Lexicographic Relevance: Selecting Information From Corpus Evidence. *International Journal of Lexicography* 16(3):251–280.

Baader, Franz, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. 2003. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press.

Babko-Malaya, Olga, Ann Bies, Ann Taylor, Szuting Yi, Martha Palmer, Mitch Marcus, Seth Kulick, and Libin Shen. 2006. Issues in Synchronizing the English Treebank and PropBank. In *Proceedings of the COLING/ACL Workshop on Frontiers in Linguistically Annotated Corpora*, pages 70–77. Sydney, Australia.

Baumgartner, Peter and Aljoscha Burchardt. 2004. Logic Programming Infrastructure for Inferences on FrameNet. In *Proceedings of the 9th European Conference on Logics in Artificial Intelligence*, pages 591–603. Lisbon, Portugal.

Bird, Steven and Mark Liberman. 2001. A Formal Framework for Linguistic Annotation. *Speech Communication* 33(1,2):23–60.

Boas, Hans C. 2005. Semantic Frames as Interlingual Representations for Multilingual Lexical Databases. *International Journal of Lexicography* 18(4):445–478.

Brants, Sabine, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith. 2002. The TIGER Treebank. In *Proceedings of the Second Workshop on Treebanks and Linguistic Theories*, pages 24–41. Sozopol, Bulgaria.

Broekstra, Jeen, Arjohn Kampman, and Frank van Harmelen. 2002. Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema. In *Proceedings of the First International Semantic Web Conference*, vol. 2342, pages 54–68.

Burchardt, Aljoscha, Katrin Erk, Anette Frank, Andrea Kowalski, Sebastian Padó, and Manfred Pinkal. 2006. The SALSA Corpus: a German Corpus Resource for Lexical Semantics. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, pages 969–974. Genoa, Italy.

Carletta, Jean, Stefan Evert, Ulrich Heid, Jonathan Kilgour, Judy Robertson, and Holger Voormann. 2003. The NITE XML Toolkit: Flexible Annotation for Multi-modal Language Data. *Behavior Research Methods, Instruments, and Computers* 35(3):353–363.

Carreras, Xavier and Lluís Màrquez. 2005. Introduction to the CoNLL 2005 Shared Task: Semantic Role Labelling. In *Proceedings of the 9th Conference on Computational Natural Language Learning*, pages 152–164. Ann Arbor, MI.

Daelemans, Walter, Antal Van Den Bosch, and Jakub Zavrel. 1999. Forgetting Exceptions is Harmful in Language Learning. *Journal of Machine Learning* 34(1-3):11–41.

Dickinson, Markus. 2005. Rule Equivalence for Error Detection. In *Proceedings of the 5th Workshop on Treebanks and Linguistic Theories*, pages 187–198. Prague, Czech Republic.

Eckart, Richard and Elke Teich. 2007. An XML-based Data Model for Flexible Representation and Query of Linguistically Interpreted Corpora. In G. Rehm, A. Witt, and L. Lemnitzer, eds., *Data Structures for Linguistic Resources and Applications*, pages 327–336. Tübingen, Germany: Gunter Narr Verlag.

Fillmore, Charles J. 1985. Frames and the Semantics of Understanding. *Quaderni di Semantica* IV(2):222–254.

Fillmore, Charles J., Christopher R. Johnson, and Miriam R.L. Petruck. 2003. Background to FrameNet. *International Journal of Lexicography* 16:235–250.

Francopoulo, Gil, Monte George, Nicoletta Calzolari, Monica Monachini, Nuria Bel, Mandy Pet, and Claudia Soria. 2006. LMF for Multilingual, Specialized Lexicons. In P. Zweigbaum, S. Schulz, and P. Ruch, eds., *Proceedings of the LREC Workshop on Acquiring and Representing Multilingual, Specialized Lexicons*, pages 27–32. Genoa, Italy.

Frank, Anette. 2004. Generalisations over Corpus-induced Frame Assignment Rules. In C. J. Fillmore, M. Pinkal, C. F. Baker, and K. Erk, eds., *Proceedings of the LREC Workshop on Building Lexical Resources From Semantically Annotated Corpora*, pages 31–38. Lisbon, Portugal.

Frank, Anette, Hans-Ulrich Krieger, Feiyu Xu, Hans Uszkoreit, Berthold Crysmann, Brigitte Jörg, and Ulrich Schäfer. 2007. Question Answering from Structured Knowledge Sources. *Journal of Applied Logic* 5(1):20–48.

Gildea, Daniel and Daniel Jurafsky. 2002. Automatic Labeling of Semantic Roles. *Journal of Computational Linguistics* 28(3):245–288.

Grosof, Benjamin N., Ian Horrocks, Raphael Volz, and Stefan Decker. 2003. Description Logic Programs: Combining Logic Programs with Description Logic. In *Proceedings of the Twelfth International World Wide Web Conference*, pages 48–57.

Guo, Yuanbo, Zhengxiang Pan, and Jeff Heflin. 2005. LUBM: A Benchmark for OWL Knowledge Base Systems. *Journal of Web Semantics* 3(2):158–182.

Haase, Peter, Jeen Broekstra, Andreas Eberhart, and Raphael Volz. 2004. A Comparison of RDF Query Languages. In *Proceedings of the Third International Semantic Web Conference*, pages 502–517.

Hajičová, Eva. 1998. Prague Dependency Treebank: From Analytic to Tectogrammatical Annotation. In *Proceedings of the First Workshop on Text, Speech and Dialogue*, pages 45–50. Brno, Czech Republic.

Hanks, Patrick. 2000. Do Word Meanings Exist? *Computers and the Humanities* 34(1-2):205–215. Special Issue on SENSEVAL.

Heid, Ulrich, Holger Voormann, Jan-Torsten Milde, Ulrike Gut, Katrin Erk, and Sebastian Padó. 2004. Querying both Time-aligned and Hierarchical

Corpora with NXT Search. In *Proceedings of LREC-2004*, pages 1455–1459. Lisbon, Portugal.

Hepple, Mark and Josef van Genabith. 2000. Experiments in Structure-Preserving Grammar Compaction. In *First Meeting on Speech Technology Transfer*. Seville, Spain.

Hovy, Eduard, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: The 90% solution. In *Proceedings of the joint Human Language Technology Conference and Annual Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 57–60. New York City, NY.

Ide, Nancy and Keith Suderman. 2007. GrAF: A Graph-based Format for Linguistic Annotations. In *Proceedings of the ACL Workshop on Linguistic Annotation*, pages 1–8. Prague, Czech Republic.

Kepser, Stephan. 2004. A Simple Proof of the Turing-Completeness of XSLT and XQuery. In *Proceedings of Extreme Markup Languages*. Montreal, QC.

Kilgarriff, Adam. 1997a. I Don't Believe in Word Senses. *Computers and the Humanities* 31:91–113.

Kilgarriff, Adam. 1997b. Putting Frequencies in the Dictionary. *International Journal of Lexicography* 10(2):135–155.

Kilgarriff, Adam and Joseph Rosenzweig. 2000. Framework and Results for English SENSEVAL. *Computers and the Humanities* 34(1-2):15–48.

Klein, Dan and Christopher D. Manning. 2003. Accurate Unlexicalized Parsing. In *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pages 423–430. Sapporo, Japan.

Knublauch, Holger, Mark A. Musen, and Alan L. Rector. 2004. Editing Description Logic Ontologies with the Protégé OWL Plugin. In *Proceedings of the International Workshop on Description Logics*, pages 70–78. Whistler, BC.

Lai, Catherine and Steven Bird. 2004. Querying and Updating Treebanks: A Critical Survey and Requirements Analysis. In *Proceedings of the Australasian Language Technology Workshop*, pages 139–146. Sydney, Australia.

Laprun, Christophe, Jonathan Fiscus, John Garofolo, and Sylvain Pajot. 2002. Recent Improvements to the ATLAS Architecture. In *Proceedings of the Second Human Language Technology Conference*, pages 263–268. San Diego, CA.

Marcus, Mitchell, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Journal of Computational Linguistics* 19(2):313–330.

Miltsakaki, Eleni, Nikhil Dinesh, Rashmi Prasad, Aravind Joshi, and Bonnie Webber. 2005. Experiments on Sense Annotations and Sense Disambiguation of Discourse Connectives. In *Proceedings of the 5th Workshop on Treebanks and Linguistic Theories*. Barcelona, Spain.

Narayanan, Srini, Charles J. Fillmore, Collin F. Baker, and Miriam Petruck. 2002. FrameNet Meets the Semantic Web: A DAML+OIL Frame Representation. In *Proceedings of the AAAI Workshop on Semantic Web Meets Language Resources*. Edmonton, Canada.

Oliva, Karel. 2001. The Possibilities of Automatic Detection/Correction of Errors in Tagged Corpora: a Pilot Study on a German Corpus. In *Proceedings of the 4th Workshop on Text, Speech, and Dialogue*, pages 39–46. Zelezna Ruda, Czech Republic.

Palmer, Martha, Dan Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Journal of Computational Linguistics* 31(1):71–106.

Polguère, Alain. 2006. Structural Properties of Lexical Systems: Monolingual and Multilingual Perspectives. In *Proceedings of the COLING/ACL Workshop on Multilingual Language Resources and Interoperability*, pages 50–59. Sydney, Australia.

Pradhan, Sameer, Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2007. OntoNotes: A unified relational semantic representation. In *Proceedings of the First IEEE International Conference on Semantic Computing*, pages 517–526. Irvine, CA.

Renear, Allen, David Dubin, and C.M. Sperberg-McQueen. 2002. Towards a Semantics for XML Markup. In *Proceedings of the ACM Symposium on Document Engineering*, pages 119–126. McLean, VA.

Romary, Laurent and Nancy Ide. 2004. International Standard for a Linguistic Annotation Framework. *Journal of Natural Language Engineering* 10(3-4):211–225.

Scheffczyk, Jan, Collin F. Baker, and Srini Narayanan. 2006. Ontology-based Reasoning about Lexical Resources. In *Proceedings of the 5th Workshop on Interfacing Ontologies and Lexical Resources for Semantic Web Technologies*, pages 1–8. Genoa, Italy.

Spohr, Dennis and Ulrich Heid. 2006. Modeling Monolingual and Bilingual Collocation Dictionaries in Description Logics. In *Proceedings of the EACL Workshop on Multiword Expressions in a Multilingual Context*, pages 65–72. Trento, Italy.

Stede, Manfred. 2004. The Potsdam Commentary Corpus. In *Proceedings of the ACL Workshop on Discourse Annotation*, pages 96–102. Barcelona, Spain.

Thompson, Henry S. and David McKelvie. 1997. A Hyperlink Semantics for Standoff Markup of Read-only Documents. In *Proceedings of SGML Europe*, pages 227–229. Barcelona, Spain.

Trippel, Thorsten. 2006. *The Lexicon Graph Model: A Generic Model for Multimodal Lexicon Development*. Saarbrücken, Germany: AQ-Verlag.