Undefined 00 (2012) 00
IOS Press

Natural Language Generation and Semantic Web Technologies

Editor(s): Name Surname, University, Country
Solicited review(s): Name Surname, University, Country
Open review(s): Name Surname, University, Country

Nadjet Bouayad-Agha [a,*], Gerard Casamayor [a] and Leo Wanner [a,b]
[a] *Department of Information and Communication Technologies, Pompeu Fabra University, C/ Roc Boronat, 138, 08018 Barcelona, Spain*
[b] *Catalan Institute for Research and Advanced Studies, Passeig Lluís Companys, 23, 08010 Barcelona, Spain*
E-mail: {nadjet.bouayad|gerard.casamayor|leo.wanner}@upf.edu

Abstract. Natural Language Generation (NLG) is concerned with transforming some formal content input into a natural language output, given some communicative goal. Although this input has taken many forms and representations over the years, it is the semantic/conceptual representations that have always been considered as the "natural" starting ground for NLG. Therefore, it is natural that the semantic web with its machine-processable semantic data paradigm has attracted the interest of NLG practitioners from early on. We attempt to provide an in-depth overview of the approaches to NLG from semantic web data, emphasizing where robust, sustainable techniques have been used and pointing out weaknesses that still need to be addressed in order to improve both the performance and scalability of semantic web NLG.

Keywords: semantics, natural language text generation, semantic web formalisms, web resources

1. Introduction

Natural Language Generation (NLG) is concerned with transforming a given formal content input into a natural language output, given some communicative goal in a specific context [108]. This input has taken many forms and representations over the years, from linguistic surface-oriented structures over semantic or conceptual representations to raw numeric data. However, it is the semantic/conceptual representations that have always been considered to be the "natural" starting ground for NLG: linguistic surface-oriented structures already predetermine the linguistic form of the output beforehand, which is clearly undesirable for flexible NLG, and raw numeric data require prior preprocessing that is not related to NLG. Therefore, it is not surprising that the semantic web (SW) with its *machine-processable semantic data* paradigm has attracted the interest of NLG practitioners from early on. The objective of this article is to provide an in-depth overview of the approaches to NLG from SW-data and the use of NLG in the SW-context, emphasizing where robust, sustainable techniques have been used and pointing out weaknesses that still need to be addressed in order to improve both the performance and scalability of semantic NLG.

We begin with a brief overview of NLG that delimits its scope, introduces its key tasks, challenges and summarizes the current state of the art (Section 2). Next, we discuss the key issues that we think make generation that uses SW-technology/data different from NLG that draws upon more traditional semantic representations (Section 3). We then review NLG research that exploits SW-data (Section 4) from two different angles—NLG for ontology engineering vs. NLG for publication of the content modelled by the ontologies.

*Corresponding author

In Section 5, we summarize what we consider the most prominent “burning” issues for a successful symbiosis between NLG and SW, before concluding in Section 6.

2. A brief overview of NLG

Fully-fledged NLG implies a number of tasks. The five most central of them are: 1) *content selection* that determines which parts of the content received as input are to be verbalized according to the context; 2) *discourse and information structuring* that ensures that the content is rendered as a coherent narration; 3) *aggregation* that merges partially overlapping content and linguistic structures to avoid repetition and to improve the fluency of the output; 4) *lexicalization* that maps conceptual (or language-independent semantic) configurations onto language-specific semantemes and lexemes, including what is commonly referred to as *generation of referring expressions* (GRE), i.e., generation of anaphora and generation of references to entities supposedly already present in the reader’s world model; and 5) *morpho-syntactic realization* and *linearization* that deals with the projection of the discourse or sentence plan obtained from the preceding tasks onto the surface.

In order to introduce the reader into the context of SW-technologies in NLG, let us give a brief overview of NLG in general and of NLG-tasks that are mainly concerned with semantics (namely content selection, discourse structuring, and lexicalization) in particular.

2.1. A bird’s eye view of NLG

As already mentioned above, the global task of NLG is to map a given formal input onto a natural language output, with the objective to achieve a given communicative goal in a specific context. The context can be entirely implicit if the generator focuses on one specific type of report for one specific type of user (as, e.g., in the case of the generation of clinical narratives for medical personnel), or allow for an explicit parameterization of only one or several features (as, e.g., in the case of the generation of explanations for either laymen or experts of a domain, or in the case of the generation of commentaries for various types of users, varying information needs and various communicative goals). As a rule, individual generators do not cover several genres.

The input can be further characterized with respect to its type (i.e., whether it is structured or not, and what language representation it uses, if any), size, domain and task (in)dependence; the output with respect to its size, coherence, fluency, language and modality; and the context with respect to the profile and information need of the target audience and the communicative goal of the generator. See Table 1 for a summary of these characteristics.

The range of admissible (or desired) characteristics of the input, output and context determines, to a certain extent, the architecture of the generator. Thus, a generator that accepts as input small unstructured sets of data and generates out of them short monolingual messages will have a simpler architecture than a generator that takes as input large semantic graphs to generate multilingual texts that vary in content, language and style according to the user profile and request. The format of the input may also vary, depending on whether a generator is used as a stand-alone application or is part of a larger automatic information processing application such as Dialogue, Question Answering, Summarization, or Machine Translation. This highlights the decisive difference between NLG and, for instance, parsing: NLG cannot always start from the same input (while parsing always starts from the language surface).[1] The consequence of this difference is that in NLG-research no consensus has been achieved so far on what a generation application is supposed to start from and what the standard input representation for generation should look like exactly—although it seems clear that “some sort of” semantic representation is the most appropriate starting ground. Over the years, different types of semantic representations have been experimented with—including model-theoretical semantics [6,30,101], Schank’s scripts [66,94], Sowa’s conceptual graphs [99], a variety of frame representations such as KL-ONE and LOOM [65,72,106], up to semantic web representations; see the following sections. Cf. also [114,119] for somewhat outdated detailed reviews.

The methodologies applied in NLG to map the given input onto the natural language output range from the use of simple fill-in templates and canned text for straightforward verbalization of messages of limited complexity to the exploitation of strategies that implement informed projections between theoretically-sound representations for each individual generation

[1]To illustrate this problem, a famous statement by Yorick Wilks that “the difference between Natural Language Understanding and Natural Language Generation is like the difference between counting from one to infinity and from infinity to one” is often quoted.

Input

Type	structured input data representation (e.g., semantic graph, database) or unstructured input representation (e.g., tabular, template); input representation language (first-order logic, OWL-DL, etc).
Size	small (e.g., a small RDF graph), large or very large input (e.g., hundreds of thousands of measurements or database entries).
Domain independence	input representation domain-independent (e.g., conceptual representation) or domain-dependent
Task independence	input representation independent or dependent of the task of text generation.

Context

Targeted genre	term definition, report, commentary, narrative, etc.
Targeted audience	lay person, informed user, domain expert, etc.
Request	information solicitation, decision support request, etc.
Communicative goal	exhaustive information on a theme, advice, persuasion, etc.
User profile	user preferences, needs or interests in the topic, individual expertise, previous knowledge, discourse history, etc.

Output

Size	single sentence, paragraph or a multi-paragraph text
Coherence	set of disconnected sentences or a coherent paragraph
Fluency	fluent NL, controlled NL, or telegraphic style
Language	monolingual (English, French, German, Spanish, . . .) or multilingual
Modality	textual only (written or spoken) or multimodal (e.g., text or speech with table or figures) and the degree of the multimodality

Table 1

Summary of dimensions of NLG-system input, output and context

task. However, as van Deemter et al. [44] point out, the distinction between practical application-oriented template-based NLG and "real" research NLG is becoming increasingly blurred in that template-based generation becomes quite sophisticated and research-oriented NLG-experiments often bridge tasks that are not in focus by simple realizations or omit them altogether. As a consequence, the theoretical justification, maintainability, and output quality and variability cannot be prognosticated only based on the fact whether templates are used or not. Until recently, rule-based realizations prevailed for methodologies of all complexities. This is about to change. Statistical realizations of different NLG-tasks are increasingly popular [122].

The generation tasks, although conveniently separated (see above), are not independent of each other and can be staggered, in the best of cases, in the generation architecture [113]. For instance, the task of lexicalization is influenced by content selection, discourse structuring, syntacticization, etc.; aggregation is to be controlled at the conceptual level, language-dependent semantic level and syntactic level; (surface) syntactic structure determination depends on lexicalization, information structure, etc.

Especially in the earlier days of generation, various architecture models have been experimented with to accommodate best for these dependencies. In the course of the years, a model has emerged that does not conform to the ideal theoretical vision of generation because it does not reflect the interdependencies between the different tasks, but which is sufficient as a working architecture. It consists of three main pipelined modules [91,107]: 1) *document or text planning* (sometimes also referred to as *content determination* or *macro-planning*), 2) *sentence planning* (otherwise known as *micro-planning* in opposition to the previous macro-planning module), and 3) *surface realization*. The document planning module is in charge of deciding *what to say* and organizing the chosen content into a coherent whole. The sentence planner is in charge of mapping the text plan to the linguistic structure, grouping information into sentences and performing aggregation, and lexicalization along the way. Finally, the surface realizer is responsible for rendering each sentence plan into a sentence string. It is obviously the document planning module and partially also the sentence planning module that must be able to cope with semantic representations.

The evaluation of the performance of each module and of the individual tasks within each module has been increasingly given more prominence in NLG. As for any NLP-application, this performance can be assessed either from a qualitative or a quantitative angle. For "deeper" tasks such as document planning, so far nearly exclusively qualitative evaluation has been applied in that human judges were asked to rate the appropriateness of the content and discourse structure in automatically generated texts. However, quantitative evaluation of at least content selection, i.e., evaluation that uses statistical measures to assess the concord of the output of the content selection task with a *gold standard*, is becoming relevant on the research agenda of NLG; see, for instance, [22] for the description of the RDF-content selection shared task challenge. The success of the standardization of content selection strategies and their evaluation will certainly also depend on the prominence of the SW-technologies in NLG. For linguistic (or "more surface-oriented") tasks such as syntactic generation, increasingly quantitative evaluation is used [11,12,14]—although qualitative evaluation is still popular [13].

2.2. *Semantically-oriented NLG-tasks*

The design and realization of nearly each of the generation tasks listed at the beginning of Section 2 depend on the type of the semantic input structure used by the generator, especially content selection, discourse structuring, and lexicalization since they operate directly on the input structure or on a fragment of it. The first two tend to output the same type of semantic structure as they take as input, while the third (lexicalization) tends to output a lexicalized structure in which the items differ in type from the semantic structure. Let us discuss each of these three tasks in turn.

2.2.1. *Content selection*

Often, an NLG-application has to address the problem of selecting a subset of content from a large semantic network, i.e., do content selection. This can be done in terms of rule-based templates, resorting, for instance, to SPARQL queries, as, e.g., in [24,27] (adopting the so-called "closed planning") or exploiting the network topology to perform an informed search of the most relevant nodes, as, e.g., in [45,100] (adopting the so-called "open planning"). Open planning is especially favoured in bottom-up, data-driven approaches where the communicative goal is to "say everything there is to say about an object of the domain", as opposed to top-down approaches where content selection is guided by explicit communicative goals which must be satisfied.

In approaches that follow the open planning paradigm, content is often seen as forming a content graph where nodes correspond to content atoms (e.g., facts in a KB or database cells), while edges indicate selection constraints between pairs of nodes. In some cases, the selection constraints are derived from links between data found in the content. The links serve as indicators of related content that can be selected together. In other cases, constraints are elicited from special relations which indicate potential discourse relations between facts when placed in a discourse plan. Thus, potential rhetorical relations are established between sets of facts in O'Donnell et al's [100] ILEX system, while in the work by Demir et al. [45] *attractor* and *repeller* relations indicate discourse compatibility (or incompatibility) between facts.

The nodes and edges of the content graph on which open planning strategies operate can be assigned weights that modulate the strength of the constraints or quantify the degree of interest of the user for certain types of content, as encoded in a user model. Weights warrant the application of optimization- and graph-based algorithms to solve the content selection problem. They can be assigned either manually as in [45] or be statistically inferred from a corpus of texts aligned with data [5,25].

A number of proposals for open planning content selection (see, e.g., [89,90]) have been realized explicitly using SW-representations; see Section 4.

2.2.2. *Discourse structuring*

A number of generators handle content selection and discourse structuring as one task [108], using text schemata in the sense of [87]. However, theoretically it is undisputed that discourse structuring is an NLG-task on its own. Discourse structuring is concerned with the derivation of a coherent discourse structure of the content that is to be turned into natural language. In this context, a very popular discourse structure theory is the *Rhetorical Structure Theory* (RST) [82] because of its pre-realizational definition of rhetorical relations in terms of speaker's intentions and effects on the hearer on the one hand, and the distinction between the main (nucleus) and supporting (satellite) arguments of the asymmetric discourse relations on the other hand.

Discourse structuring can be carried out either after content selection (as, e.g., in [67,83,109]) or interleaved with content selection (as, e.g., in [85,96,100]), such that during the relevant content search through the graph only those nodes are taken into account that can be connected to the already selected nodes via a rhetorical relation—which ensures discourse coherence. The rhetorical relations between content nodes are introduced either directly into the content graph prior to NLG-proper (as, e.g., in [109]), via *plan operators* during text planning (as, e.g., in [67,85,96]) or via a projection from semantic relations established between content nodes [24,76].

It has been argued by Marcu [83] that the asymmetric discourse relations (such as ELABORATION, CAUSE, and IMPLICATION) reveal a *canonical* ordering. In RST-terms, this means that the nucleus comes before the satellite. This, together with other local coherence constraints such as adjacency, can be exploited to achieve global coherence—as done by Marcu in his constraint-based approach.

Another issue that is relevant to discourse structuring is how to package units of content into *Elementary Discourse Units* (EDUs). This issue has been addressed, for instance, using templates in a pipeline architecture (as, e.g., [23,24]) in order to ensure that each EDU can be rendered in natural language, and in terms of a topological search in an iterative NLG-architecture (as, e.g., [35]); see also Section 4.2.

2.2.3. Lexicalization

Rather different strategies have been discussed to realize the mapping between the semantic and lexical entities: discrimination networks [57], semanteme-lexeme association with additional variation [75,79, 102,120], semantic vs. lexical feature or structure matching [98] or unification [49,99,114], etc. However, some proposals also treat the items of the semantic representation as lexical items, such that they do not change the type of the output; see e.g., [106] and, more recently, [116].

To facilitate the mapping of a semantic (or, more precisely, language-independent) representation onto an abstract linguistically-oriented representation that gives sufficient room for flexible verbalization, language-oriented ontologies have been introduced into generation. The most prominent of them has been, without doubt, the *Penman Upper Model* (UM) [7], which bridged the gap between LOOM representations and linguistic structures. Originally used in the context of systemic-functional generators PENMAN [84] and KMPL [9], the UM evolved over the years into a major multilingual linguistically-oriented ontology known as the *Generalized Upper Model* (GUM); see, for instance, [10]. GUM is available in OWL-DL (see also Section 5).

3. Semantic web representations: What makes the difference?

In the previous section, we saw that semantics plays a crucial role in various tasks of NLG. In this section, we discuss the key issues that make NLG that uses SW-technology and/or data different from NLG that draws upon more traditional semantic representations. These key issues concern (i) the uniform codification of NLG-relevant information in standard, freely available SW language formalisms; (ii) the use of SW technologies for reasoning and accessing domain and NLG-specific knowledge; (iii) the scalability of the NLG-techniques that are able to cope with large scale SW repositories; (iv) the portability of generators that are based on SW technologies; and (v) the reuse of SW resources across generators.

3.1. Uniform codification of NLG-relevant information

One of the central problems in NLG is the projection of domain information structures onto language-oriented structures that render this information. Obviously, this problem is simplified considerably if for both types of structures the same formal representation is used, or if the domain information structures are enriched by the information how to project them onto language-oriented structures.

The use of a single knowledge base (KB) that contains information about both the domain and the projection of the domain onto language was already common practice in traditional knowledge-based NLG. However, these KBs were, as a rule, created deliberately for specific applications, often with no clear separation between domain knowledge and generation task-specific knowledge—which raised questions on the theoretical soundness and portability of such generators. As a consequence, the need for a separation between different types of knowledge was recognized in several works predating the appearance of SW-standards and a separation of knowledge into different but uniformly coded layers has been suggested. For instance, Fröhlich and Van de Riet [53] implemented in LOOM [81] a three layer ontology consisting of a domain ontology, a lexical ontology and an upper model.

OWL/RDF offers explicit support for the modularization of knowledge into different ontologies. This support has been exploited to model task-specific and linguistic knowledge. The knowledge represented in terms of OWL-ontologies ranges from content schemas, ordering, discourse relations, plan operators, models and profiles of the user to lexical knowledge. Thus, in accordance with Kittredge et al.'s [75] argumentation for the need of *domain communication* *knowledge*, Bouayad-Agha et al. [24] distinguish between three layers of ontological knowledge: domain, domain communication and communication knowledge. The domain communication layer adds, using reasoning, new domain content that is relevant to NLG: user-tailored content, qualitative views on quantitative domain data, temporal aggregation of data useful to generate temporal expressions, etc. The communication layer is a language-independent linguistic ontology that models NLG-concepts and relations instantiated during content selection and communicative/discourse structuring—among them, content schemas, discourse relations and sentences.

In Janzen and Maas's dialogue system [68], NLG starts from a layered OWL-DL knowledge repository that contains a complete model about question answering and an instantiation of the domain model (in their case, in-store shopping dialogues). The layers separate the description of the domain from the model of the task, which models in a library of schemas of questions a taxonomy of discourse intentions and user-driven composition of a set of rhetorical plan operators that specify the answers that the system can produce.

Dongilli [47] and Galanis et al. [55] use ordering annotations in the domain ontology to indicate a partial ordering, which is extended to a total ordering of the facts to be communicated in the text.

Parts of the content can also be assigned user preferences—as, for instance, done in [36,55], where the user preferences are encoded in terms of numerical weights [36,55] used when navigating through the content graph during text planning.

A complete ontological model of the user profile is used by Bouayad-Agha et al. [24] for the determination of content relevant to the user prior to NLG-proper in the context of generation of environmental information. The profile includes the user's age, activity, sensitiveness (e.g., to birch pollen) and diseases (e.g., asthma) related to environmental conditions.

The uniform codification of domain data and lexical information has received particular attention. Thus, instead of keeping the lexicon separate from the domain data, some approaches opt for annotating the data with lexical information using the same SW formalisms as for the domain data. This is, for instance, the case of NaturalOWL [55], where classes and individuals in the OWL-ontology are associated with noun phrases together with the gender of the head nouns and their singular and plural forms. Properties are assigned micro-plans for sentence planning. These micro-plans define an abstract clause structure with a verb as its head and information about the verb's inflection and valency. For the purpose of annotating ontologies, the authors of NaturalOWL developed a tool [1,17,56] supported by reasoning that can be used to assign multilingual annotations to OWL ontologies.

There have been some proposals to standardize the codification of lexical information in the SW. Thus, LingInfo [29] is an ontological model for annotating the morphosyntactic structure (i.e., inflection and decomposition) of labels in ontologies. LexOnto [32] is another model which focuses on the specification of the mapping of the semantic arguments of OWL properties to the arguments of a predicate. Buitelaar et al. [28] propose a model called *LexInfo* that unifies LingInfo and LexOnto and builds on the Lexical Markup Framework [52], a meta-model for representation of computational lexical information, adapting it to OWL/RDF notation.

In [86], a more general model called *Lemon* is presented. Lemon makes no commitment to any particular linguistic information. Rather, it serves as a modelling language to describe a wide range of computational lexicons. The authors also present a rewritten version of LexInfo in terms of Lemon.

Beyond elaborated proposals on the uniform codification of domain and lexical information, it is increasingly popular to consider the labels of entities in SW-ontologies to be largely natural language words. Thus, having analyzed the naming patterns of properties in a collection of freely available ontologies, [93,105] argue that the designers of ontologies tend to use meaningful words as labels.

This linguistic information can be used directly in the natural language output (although at the cost of low fluency; see [69]) or be exploited to reduce the cost of creating domain and ontology-specific lexicons; see, e.g., [20]. More elaborate approaches use on-line lexical resources such as WordNet or FrameNet to identify patterns in property names in ontologies and associate them with valency information; see [117] for the use of WordNet and [37] for the use of FrameNet.

Being appealing from the perspective of the development of domain-independent NLG, automatic lexicon derivation should nonetheless be accompanied by a revision by language experts to ensure a certain quality [20], or be only used as a fallback if no manual annotation is present in the ontology for a given property/concept [110,115,126]. On the other hand, the necessity for manual revision could be avoided by imposing naming conventions that restrict the grammatical category and composition of terms when authoring ontologies [62,103].

3.2. Use of SW technologies for reasoning in NLG

Reasoning is one of the most prominent advantages of adopting one of the OWL sublanguages as a description language for a domain. Many NLG-systems using SW-representations are supported by off-the-shelf reasoners that perform standard reasoning operations such as consistency checks and instance classification. For instance, the RacerPro reasoner is used to support both the Query Tool [50,51] and ELEON [17] authoring tools. The linguistically-motivated inference with the purpose to find subsuming concepts of a concept that is to be described, i.e., do content selection, as described in [89,90,92] is grounded in the RacerPro reasoner.

When the goal is to generate fluent natural language, domain ontologies do not model, as a rule, all the concepts that arise in human communication. Therefore, some systems rely on inference based on domain-specific rules in order to extend domain data and thus account for the missing concepts. See, for instance, [68] and [25,26], who employ SWRL and Jena rules respectively.

Closely related to reasoning is the support of specific OWL-sublanguages (or *profile* in OWL 2.0 terminology) in NLG-systems. Each sublanguage seeks a balance between expressivity and computational guarantees for reasoners. For instance, RacerPro requires the OWL-DL sublanguage, such that the input to the aforementioned NLG-systems is restricted to that sublanguage or to less expressive languages.

The SPARQL RDF query language has been the focus in the context of Controlled Natural Languages (CNLs) and NL-interfaces that support ontology querying in natural language. Most of the CNL and NL-interface approaches call for parsing or information extraction techniques and grammars for mapping natural language to a query language (see, e.g., [16,74,118]) and less for NLG. However, NLG has been used, for instance, for conceptual authoring and verbalization of non-standard conjunctive queries [47] as well as for verbalizing the query potential of an ontology in a non-standard query language [2].

SPARQL-queries are also the natural choice for text planning implementations that process OWL-data [3, 24,27,40,68,97], especially if they also output a content/text plan in the OWL representation language (as done, e.g., in [24,68]). Thus, in [68] already cited in Section 3.1, complex NLP-oriented SPARQL-requests associated to their plan operators are used to retrieve information from the domain KB. In [24] also cited in Section 3.1, SPARQL-queries encode both templates for text planning and the constraints governing the application of the templates. The use of SPARQL allows them to implement a text planning module that works natively on SW-technologies, with both the input data and the contextual information (i.e. user profile and query) in OWL/RDF and processing the content exclusively through SPARQL-queries. Murray et al. [97] also use SPARQL-queries to encode the high-level schemata of the text planning module. Bouttaz et al. [27] not only encode contextual knowledge in OWL and RDF, but also use SPIN-SPARQL queries to store and apply complex content selection rules.

3.3. Scaling up to large SW repositories of data

Semantic representations based on frames (such as KL-ONE or LOOM) or “home-made” idiosyncratic semantic representations common in NLG in the past, do not scale up in that they are not powerful enough to express all required meaning constellations, are too complex in maintenance or too slow in access when beyond a certain size. Semantic Web offers the chance to scale up NLG-applications to large repositories of data. The main challenge in this context is that, first of all, content selection strategies can deal with SW-repositories. Open planning-oriented content selection such as the one proposed by O’Donnell et al. [100] is easily extendible to cope with large SW-data repositories. As a matter of fact, it was extended by [55] in an OWL-ontology about museum artifacts (see Section 4.2). Dai et al. [35]’s criteria for selecting a node in an SW-network are more basic than either [100] or [45]. They rely on the notion of distance and relevance (in their terminology: whether a node is “convincing" enough to be selected), which allows them to extract fragments from the semantic network and stopping when a maximum distance from the node of interest is reached (see section 4.2). Mellish and Pan [89,90] also employ an opportunistic search for selecting and ordering subsumers of a class in an OWL-ontology using inference between axioms directed by communication constraints (see Section 4.1).

Bouayad-Agha et al. cope with large scale SW repositories in content selection by preselecting first all concepts that are related to the relevant topic according to a set of predefined rules/templates [23] or by compiling on the fly a KB based on the user request out of a large repository [24] before performing the actual content selection.

3.4. *Portability of SW-based generators to new applications*

SW-data is available for many different domains, is increasingly standardized, and has the potential for many different applications. For NLG, this means a chance to become portable across domains and applications. Portability of SW-based generators can be ensured by either enforcing a clear separation between general purpose domain knowledge and task-specific domain-independent knowledge, or by reducing the amount of task-specific knowledge needed to generate language.

Separation between domain knowledge and task-specific knowledge can be achieved by layering the different types of knowledge (as already mentioned in Section 3.1) and providing mechanisms for the mapping between domain- and task-specific knowledge. This latter is also facilitated by the use of Upper Models such as the GUM [10] (see Section 2) used, for instance, in [47].

In some cases, a set of domain-independent and linguistically-motivated relations is enough to substantially increase the portability of some NLG-modules. For instance, the generator in [21] maps all relations in the input ontologies onto one of four generic relations for which the surface generator has in-built support. In [20], the text and sentence planning modules operate on any ontology that contains the same four upper relations.

A popular strategy to reduce the amount of task-specific knowledge that is needed to generate language is the reduction of the generated language constructions to a controlled subset (the so-called *Controlled Natural Language*, CNL) for which an unambiguous mapping from the formal languages used in the SW can be defined. Power and Third [105] outline the assumptions usually followed for this mapping in a *Consensus Model*. According to this model, axioms are expressed by sentences specified by a grammar, one per axiom, and atomic terms involved in axioms are verbalized by entries in the lexicon. CNL-grammars are defined exclusively in terms of a formal language, i.e., OWL, and linguistic expressions. Since CNL-grammars are generic, only lexicons need to be defined for new domains. When generic grammars are paired with automatic derivation of lexicons from ontologies (see Section 3.1 above), the NLG-system becomes highly portable, albeit at the expense of the quality of the generated text.

A workaround for the lack or deficiency of naming conventions assumed for the automatic derivation of lexicons from ontologies [62,103] (see the discussion in Section 3.1) is to involve (expert) users of the NLG-system in the acquisition of linguistic resources for each domain, as is envisaged in conceptual authoring [63,103] (see Section 4 below) or in the ACE View ontology and rule editor [43], which support user's creation of new entries in the lexicon for the newly added ontology concepts.

Semantic wikis, where both domain content and the corresponding linguistic information can be acquired simultaneously from users, are another example of the application of CNL. De Coi et al. [43] present ACEWiki, a semantic wiki that works under the same principles as ACE's ontology and rule editors. Bao et al. [4] use variations of the Rabbit [60] and ACE CNLs to generate English and Chinese descriptions of ontologies maintained using a semantic wiki. The authors present a meta-model for the manual creation of ontologies with linguistic annotations used together with the CNL-grammars to verbalize the ontologies.

3.5. *Reusability of SW resources in NLG*

Generally speaking, the formalization of a domain in an ontology promotes reuse, especially if standard representations like the OWL/RDF-standards are used. For example, the modelling of the user using the *Problem Description Language* in a sub-ontology for an environmental report generation application [24] could be reused in any domain, and integrated in any NLG-application that takes the user model into account for a generation decision (e.g., content selection or lexicalization), especially if the other sources of knowledge are also modelled in ontologies. However, most NLG-applications developed so far and exploited their own ontologies from scratch, using SW-standards and technologies. Nonetheless, some have based their representations on existing models with or without adapting them. For instance, Bouttaz et al [27] generate descriptions of metadata (e.g., provenance and access permissions) about digital artifacts and processes in a virtual research environment based on an existing provenance model (see Section 4.2). Furthermore, they model the user's social context using their own extended version of the FOAF social networking vocabulary. Argüello et al. [3] present a system that generates clinical narratives whose domain model consists of eight ontologies: four are part of the healthcare communication standard HL7,[2] which includes the specification of a Clinical Document Architecture, three are related to standard clinical coding systems such as SNOMED CT,[3] and one has been introduced to facilitate the terminology binding process. Another domain in which existing ontologies are often reused and extended in NLG-applications is the museum domain, which uses the CIDOC-CRM[4] ontology, which models concepts and information in cultural heritage and museum documentation. NLG-applications based on an extension or exploitation of this model are presented in [121] and [40].

Promising from the perspective of reusability of SW data in NLG is the linked open data initiative, which promotes a set of principles for the publication of data, collectively referred to as the *Linked Data* paradigm, with the aim to ensure open access and integration of published data with existing resources in information technology applications [18] (see also Subsection 5.3). Having interlinked sources of data available to them, NLG-systems can operate directly on a global, cross-domain data space rather than start from isolated and domain-specific data sources. Damova and Dannélls [39] present an approach for the generation of museum artifact descriptions [40], which to our knowledge is so far the only NLG-application that draws upon existing open linked datasets integrated using a single so-called "reason-able view" accessible via a single SPARQL end-point [40]. They integrate both generic resources (DBPedia[5] (the RDF-version of Wikipedia), Geonames[6], a geographic database and Proton[7], and an Upper Level ontology), and specific resources (the CIDOC-CRM model, an intermediate ontology called the Museum Artifacts Ontology (MAO) and the Gothenburg City museum data).

4. NLG-applications in the light of SW

We subscribe to Bontcheva's [19,41] early view on NLG as having two roles to play in the SW: firstly, "[helping] users who are not knowledge engineers to understand and use ontologies", and, secondly, to provide "formal knowledge [...] expressed in natural language in order to produce reports, letters, etc. [that is to] present structured information in a user-friendly way". Subsequently, we distinguish between two main categories of NLG-applications related to SW. The first category captures Natural Language Interfaces (NLIs) for ontology engineering that use NLG-technologies to present the information in natural language to the user [112]. In these applications, the user is typically a domain expert who uses the NLI to author (i.e., model and/or populate), query or document (i.e. verbalize the *tbox* and/or *abox*) an ontology. In the second category of applications, the semantic representation is a means for codification of knowledge that is to be published using NLG-technologies, following a specific communicative goal.

There are, of course, other ways to cluster NLG-applications within and across these two objective-oriented categories. For instance, Gardent et al. [34] distinguish between applications for verbalizing ontologies, for querying ontologies and for authoring ontologies. The verbalization applications in their case subsume both approaches for documenting ontologies and for publishing knowledge in an end-user application.

In what follows, we present in some detail a number of works that fall into each of our two main categories. Given the importance of evaluation in NLG, we mention, whenever provided, the evaluation procedure and the evaluation outcome. We conclude the section with a summary of the individual approaches with respect to the NLG-dimensions and -tasks discussed in Section 2 and semantic web issues for NLG discussed in Section 3.

4.1. NLG for ontology engineering

NLG for ontology engineering is used by a domain expert for the *authoring* of queries, class definitions (i.e., *tbox*) or instances (i.e., *abox*) from an ontology using a supporting interface that guides the user through the authoring process, or for the more general presumably unguided *verbalization* of a (fragment of an) ontology. Such applications can be used for the comprehension or development of ontologies and the construction of queries for both domain experts and lay users. For instance, Hewlett et al. [62] speak of the following purposes of ontology verbalization: (i) the automatic documentation of concepts so that lay users who are involved in semantic annotation can understand them, (ii) making web service descriptions un-

[2] http://www.hl7.org/implement/standards/index.cfm
[3] http://www.ihtsdo.org/snomed-ct/
[4] http://www.cidoc-crm.org/
[5] http://dbpedia.org
[6] http://www.geonames.org
[7] http://proton.semanticweb.org

derstandable to users, and (iii) providing natural language descriptions of web-service policies or rules to policy developers (the latter is an application domain for [27] and [43]).

In both guided and unguided verbalization, the generated text must be unambiguous, which is why CNLs are often employed, especially when there are no requirements for generating a coherent text and when "round trip" authoring is desirable (i.e., verbalizing the ontology into CNL and translating the CNL back into ontology axioms).

Of course, not all NLIs for ontology engineering use NLG. For example, many natural language querying systems straightforwardly rely on parsing the natural language input entered by the user into a bag of keywords or a full syntactic tree before mapping it to a logical representation and translating it into a query; see, e.g., [74,118]. Likewise, many ontology authoring tools do not rely on NLG.[8] Some, as, e.g., [78,111], rely on a look-ahead grammar-based text editor as in the PENG-D and AceWiki authoring tools; others (such as, e.g., [46]) use a text editor that provides feedback to the domain expert about the parsed structure with respect to the recognized CNL-patterns and with respect to possible ambiguities in the ROO-authoring tool for the Rabbit-CNL. In fact, the more stilted and disconnected the output text can be, the more primitive (if used at all) the NLG- technology is, whilst the more complex, lengthy and coherent an output text is needed, the more needed and sophisticated is the NLG-technology (e.g., including aggregation and referring expression generation, grouping into paragraphs, and use of realization grammars). Unfortunately, only a few usability studies contrast different NLIs, and those that do tend to be between systems that use natural language and systems that do not [73], rather than contrasting the underlying technology of the used NL-systems (i.e., NLG vs. not-NLG).

In what follows, we present the use of NLG for unguided ontology verbalization and for interactive guided verbalization, i.e., conceptual authoring.

4.1.1. *Ontology verbalization*

Approaches to ontology or query verbalization strive for domain independence, that is, the ability to verbalize any ontology with a minimum effort for the adaptation of the linguistic resources. Therefore, the linguistic resources (if any) are limited and tend to be derived automatically from the linguistic information available in the ontology.

Hewlett et al. [62]'s aim is to generate fluent natural language descriptions. They introduce, along with an ontology browser and an editor tool called SWOOP, a tool for the verbalization of OWL-class definitions. From each class definition, the tool generates a parse tree that contains the class, its properties and entities. The tree is preprocessed to eliminate redundancies and irrelevant information and then further pruned to remove relations that require a realization in a different sentence structure. Subsequently, certain idiosyncratic concept aggregation is performed (e.g., min-cardinality and max-cardinality axioms are combined using the keyword "between"). The resulting tree is mapped onto an introductory sentence and a number of bulleted items applying simple mapping rules. The authors conducted a pilot study evaluation on five classes from freely available ontologies with ten users with no background on SW. Each user was shown a single verbalization trial consisting of the same input verbalized using the SWOOP-verbalizer or verbalized using two different generic OWL natural language renderers (one of which was the Protegé's NL renderer). She was then asked to pick the one she prefers based on the correctness of the definition, readability, and clarity. According to the authors, all the users chose the output of the SWOOP-verbalizer.

In contrast to [62], Jarrar et al. [69] present a simple engineering solution to multilingual verbalization of logical theories that can be represented in a SW-ontology, with little NLG-capability and consequently only pseudo natural language output. The solution has been implemented for the verbalization of an *Object-Role Modelling* (ORM) schema within an ORM modelling tool.[9] For each language, a template file is created which expresses in XML a template for each constraint, verbalized with the class names and property names inserted *as is* (e.g., *ProducedBy* will be inserted as is for any language). To write a new template file for another language, an ORM-engineer can use an existing template file for a similar language.[10] Ten languages have been experimented with so far. The approach has been informally tested with 40 lawyers during the development of a *Customer Complaint Man-*

[8]See also [80,112] for extensive reviews of querying and authoring interfaces.

[9]ORM is a conceptual modelling approach fully formalized in first-order logic and comprehensive in its treatment of many rules and constraint types.

[10]The authors claim that only two hours are required to develop templates for a new language.

agement Ontology. The test revealed that the domain experts could check themselves that the model indeed captures their knowledge accurately, resulting in a better representation and in an increased level of trust that there is a mutual understanding between the domain expert and modeller about the subject matter.

A number of works use CNLs. Thus, Kaljurand et al. [43,71] use the highly expressive *Attempto Controlled English* (ACE). A bidirectional translation between a subset of ACE and OWL 2 is provided. The mapping from ACE-sentences to OWL is done using a the Attempto Parsing Engine (APE), which maps ACE-sentences to first-order logic, and then to XML or OWL/SWRL. For the verbalization of an OWL-axiom in terms of an ACE-sentence, first, the axiom is rewritten into a logically equivalent axiom that can be verbalized as a meaningful sentence. Next, the remaining OWL-class descriptions and properties are verbalized as nouns and passive/active verbs respectively. Complex class descriptions that use intersection, union, complementation and property restriction are verbalized using complex noun phrases that use conjoined, disjoined, negated and embedded relative clauses. Some basic realization patterns such as denoting named classes by singular countable nouns, and (object) properties by transitive verbs in their infinitive form are used. Finally, the OWL-axioms are verbalized as ACE-sentences using realization patterns or mapping rules, so as to make the resulting sentences more readable.

The ACE verbalization tool is used in a number of SW-tools [43]—for instance, in the ACE View Protegé plugin for visualization and editing of SWRL/OWL in ACE, in the ACE Reasoner RACE for theorem proving in ACE, in the ACEWiki Semantic Wiki for content representation and editing in ACE, and in the Protune framework for policy definition and enforcement which is supplied with an ACE front-end for policy definition. The ACE View plugin also allows users to visualize potential DL-queries and answers and entailed propositions, and to override the default ACE surface realization of entities.

Davis et al. [42]'s template-based generation module of a round trip authoring (ROA) system is based on CLOnE, a very simple CNL with a bidirectional mapping to only a small subset of OWL. The goal is to provide the domain expert with an initial summary of the ontology which she can subsequently edit and thus avoid the CNL learning curve. The system starts by selecting classes, instances, class properties with their domain and range, and instance properties. The text generator is configured using an XML-file that contains a number of templates. Each template consists of an input element that defines which triples should be used and an output element that specifies the phrases to be generated and the part of the triple inputs that can be used to fill in the slots. The input element can also impose coreference between triples such that aggregation of several triples into a single sentence can be done. Basic morphological inflection is provided for verb-like properties using the SimpleNLG dictionary look up library.[11] As evaluation, a study was performed that is similar to the study used for evaluating ontology CLOnE-authoring [54], where users were asked to perform some authoring tasks using either Protegé or CLOnE after having read a short introduction about ontologies and a manual on how to edit ontologies using Protegé and CLOnE and after having seen some examples. In the evaluation study of the verbalization tool, the subjects were shown the examples but not the reference manual. After each task, each user completed a System Usability Scale (SUS) questionnaire and at the end of the evaluation session a comparative questionnaire. The results showed that subjects found ROA significantly more usable and preferable than Protegé for simple ontology editing tasks. Furthermore, whereas in the previous evaluation there was a strong correlation between CLOnE-task times and Protegé-task times, they found that this correlation was significantly weakened in the new study, with ROA tasks taking 32% less time than Protegé tasks, and that there was a high correlation between high SUS scores and task times.

A few works, most notably those by Mellish et al. [89,90,116], investigate NLG-techniques for the task of ontology/query verbalization and have thus a theoretical relevance to NLG. Sun and Mellish [116] present an experimental microplanner to produce an isolated sentence from a small RDF-graph that has a maximum of ten triples and can be a priori realized in a single sentence. Their objective is twofold. Firstly, as all works presented in this section, they aim to use domain independent resources. Therefore, they start with the automatic identification of lexical (Part Of Speech) patterns implicit in most class and property names from a large corpus of ontologies. Each pattern is mapped onto one or more lexico-syntactic rules expressed in the Lexicalized TAG (LTAG) formalism [70]. Secondly, given these resources, they

[11] `http://code.google.com/p/simplenlg/`

aim to produce a cohesive sentence by making the most appropriate lexicalization choice for each triple. The cohesive production of a sentence depends on a number of constraints, which are: 1) to be syntactically consistent with similar "aggregable" triples in the neighbourhood, 2) to be compatible with the realization choices already made, and 3) to achieve an overall balanced sentence with no great discrepancy between the length of the subject Noun Phrase (NP) and the Verb Phrase (VP). The compatibility with previous realization choices is naturally taken into account in LTAG, where each minimal syntactic structure contains explicit information on how other structures can be adjoined to it. The authors present an algorithm that transforms the input RDF-graph into a tree, identifies neighbourhoods and maps them onto lexicalized representations. In order to achieve a more balanced final NP for each neighbourhood, the algorithm shifts some of the information up the lexicalized trees of some triples before aggregation of the lexicalized triples in each neighbourhood and joining all the neighbourhood-level syntactic structures is applied to form a single lexicalized syntax tree, which is then linearized into a sentence. No evaluation is presented.

Mellish and Pan [89,90] address the problem of domain-independent content determination (i.e., selecting the content and organizing it into a coherent whole) from an OWL DL ontology for *tbox* verbalization. They argue that heuristic-based navigation as used, e.g., in [100], is not adapted to the challenge of *tbox* verbalization as it leads to clumsy overly complex sentences with overlapping and possibly misleading information due to the false implicatures that can be drawn from some of the ontological axioms [92]. They propose, instead, a new approach called "Natural Language Directed Inference" (NLDI) that selects axioms if they can be inferred from previously selected axioms according to such criteria as preference of shorter formulas to long ones, realizability of selected axioms, and logical independence of the original axioms (such that they are included only once). NLDI has been implemented experimentally to give an answer to the question *What is an A?*, where A is the atomic concept in an OWL-DL ontology. The approach consists in a depth-first search of refined concepts (i.e., subsumers) of the original concept of increasing syntactic complexity, as long as the new concept is not already known to the user or until a preset complexity limit is reached. The candidates are filtered using so-called *natural language direction* rules such as introduction of more specific concepts before more general ones and evaluation of the candidate against a set complexity limit. The NLDI-implementation was tested on two small ontologies, and although being time consuming, it was able to find natural language subsumers in a large search space.

Stevens et al. [115] discuss the generation of OWL-class descriptions in natural language from a bio-ontology.[12] Like Sun and Mellish [116] (see above), they construct the lexicon automatically from the linguistic information available in the atomic entities (or from their associated ontology labels). For each class in the ontology, they first select all axioms in which the class occurs as a top-level argument. Next, axioms that share a common pattern and differ in only one constituent are aggregated. Each (aggregated) axiom is realized as a sentence using a generative grammar that has rules for nearly every logical pattern in OWL-DL. The axioms are presented in the order in which they were originally retrieved in the ontology. In an evaluation study of an improved version of their system, the authors asked potential users of the ontology how understandable the generated definitions are. They found that most users rate highly or very highly the understandability of the generated descriptions. However, this result is dampened by an additional study in which the users were provided with alternative wordings of axioms containing some properties and asked them which is the most natural to read and which best captures the meaning of the OWL-content. They found that the texts rated as most natural were also the ones least capturing the OWL-meaning, and vice versa, thus suggesting that there is a trade-off between naturalness of the output and faithfulness to the underlying representation.

Contrary to the works above, Ang et al. [2] are concerned with the verbalization of queries from an ontology. Their system, *KnowleFinder*, generates for domain experts natural language (NL) statements of all the queries that can be built from a bio-ontology given a single query term, with the goal that the domain experts understand better the conceptualization of an ontology and identify whether the ontology supports queries relevant to their needs. First, all transitive query paths from the query term and across object properties are obtained. These paths are translated into NL-statements. Each statement represents a valid query which may return some results from the OWL-DL KB made available to users via a hyperlinked URL.

[12] Related work is presented in [105,126].

The input to the NLG-component proper is a set of triples, which are mapped to a set of NL-query statements by 1) a set of mapping rules obtained by a rule learning algorithm that takes as training data user-provided examples of triples and corresponding NL statements, and 2) lexical realization of object properties as determined automatically by a text mining tool that performs PoS-tagging and term extraction. The generated NL-query statements are aggregated to form a compact query using a set of recursively applied aggregation patterns. In order to increase the grammaticality of the generated NL-query, an off-the-shelf grammar checker is used.

4.1.2. Conceptual Authoring

In conceptual authoring, a term coined by Hallet et al. [59] but elsewhere named WYSIWYM (*What You See Is What You Meant*) (starting at least from [104]), the user authors the concepts of the ontology schema to formulate a query or to design the ontology. More specifically, the user edits the underlying knowledge representation displayed to her via an interface in terms of NL-statements generated automatically from the knowledge representation. The editing is done through substitutions in specific place-holder points in the rendered text, where the list of possible substitutions presented to the user is delimited by the system according to the underlying knowledge representation. Thus, as with other NL-interfaces, there is no need to know complex query or ontological languages. The interface displays to the user the knowledge that is to be authored in natural language. The expertise and training needed for authoring is thus minimal. However, what makes conceptual authoring different from other ontology editors, including CNL-based ones, is that there is no need for language interpretation. Complex knowledge configurations can be authored without the interpretation bottleneck. In addition, in the case of querying, one can be sure that the input to the system matches a valid query, thus avoiding the pitfall of "linguistic vs. conceptual failure" [51], where the user does not know whether a query failed because no data was returned or because it was not consistent with the KB-schema.

Though initially applied to relational databases and pre-SW KBs [58,59,104], the use of conceptual authoring for SW-ontologies was a natural step forward. Thus, Power [103] describes a proof-of-concept prototype editing tool that uses WYSIWYM for authoring both the instances and definitions of a very restricted DL-ontology. In this prototype, the user is able to add new terms (property, class or individual) and specify their linguistic category, thus realizing the conceptual and linguistic open-endedness requirement for a SW-authoring tool [63] (see below). Furthermore, the tool uses the reasoning services available for DL-languages to determine and present only relevant feedback text to the user (e.g., the feedback text *Mary owns an animal* is not presented if the user already specified that *Mary owns a pet*).

Power discusses further issues involved in the development of a conceptual authoring tool for SW-ontologies, including the viability and overhead involved in letting the user specify the linguistic realization of the ontology concepts and properties, the scaling up of such an authoring to full DL and to thousands of terms and axioms, and the need for more sophisticated NLG to improve the coherence of longer feedback text using aggregation, pronominalization, discourse planning and content summarization.

Hielkema et al. [63,64] implement a metadata editing tool using WYSIWYM to allow social science researchers to deposit and describe their data in an OWL Lite ontology via a semantic graph. The semantic graph represents the knowledge presented. It is edited by the user and constantly updated after each user feedback. At the end of an interaction session, the semantic graph is translated into a set of RDF-triples and stored in a shared repository. The text planner that ensures a coherent presentation of the data operates on the semantic graph, creating HTML paragraphs and headers, one per object. Properties of the semantic graph are grouped according to their common source, their label and their target nodes (in this order). Properties with common sources or targets are marked for aggregation that is performed by the surface realizer. Each property is mapped to a dependency tree for realization. The surface realizer verbalizes the dependency trees using the SimpleNLG[13] package. In the course of the verbalization, some limited aggregation and pronominalization is carried out.

The authors evaluate their tool by measuring the performance of fifteen subjects on four successive tasks (e.g., entering research data, deposition date, access, type and topic of transcript material) and find its usability not as positive as expected. One of the reasons, they argue, is the complexity of the ontology with many options to chose from, some of which the user is not familiar with. Furthermore, they observe [63]

[13] `http://code.google.com/p/simplenlg`

that the portability of their approach is unclear because the ontology and the authoring tool influenced each other's development so that the ontology is geared towards the NLG-task. According to Hielkema et al., a requirement for a conceptual authoring tool for the SW is that it should be portable, flexible and open-ended, supporting dynamic evolution of metadata (such that the user is able to add new properties or concepts to the ontology via the conceptual authoring tool). As a mechanism for ensuring open-endedness while controlling the quality of the data entered by the user, the authors propose to integrate ontologies with folksonomies, whereby tags entered as new values for datatype properties are recorded into folskonomies so that the same tags can be suggested to the user when entering further values.

Dongilli and Franconi [48] present a work-in-progress WYSIWYM-like NLI for authoring conjunctive queries in a tree-like representation. The user can manipulate the query by, for example, generalizing/specializing a term or deleting a focused concept. Unlike WYSIWYM, Dongilli and Franconi's NLI relies on logic-based reasoning for filtering the content presented to the user and on the full-fledged surface generation platform KPML [9] for generation. The authors justify the use of KMPL by KPML's features to accept as input an ontology-oriented logical specification formulated in terms of the *Sentence Plan Language* [72], to provide large scale NLG-resources (including grammars and task- and domain-independent linguistically-motivated ontologies such as the Penman Upper Model [8]) for a number of languages, and to offer a stable grammar development framework.[14]

In Franconi et al. [50], the KPML-generator has been replaced by a template-based generator that aims to be domain-independent. To use it, a lexicon and template maps (for the projection of each concept/role name to a generation template) must be specified. Both can be partially derived from the linguistic information present in the ontology, as in approaches to ontology verbalization described in the previous subsection. According to the authors, for 65% of the relations (of a total of only 64 relations) in their test ontologies, the generated template is suitable for use in the NLI.

[14]A more complete description of the used generation architecture is given in [47].

4.2. NLG for knowledge publishing

The approaches to knowledge publishing in natural language are very heterogeneous. Some of them are application-oriented (see, e.g., et al. [121] and [40]), others (such as, e.g., [23,24,35,55] aim to provide a theoretically sound framework. In general NLG-terms, we can distinguish, for instance, between knowledge publishing as question-answering (or dialogue) vs. publishing as report generation; publishing of existing ontologies vs. publishing of ontologies harvested before from texts; publishing that involves a major content selection task vs. publishing of the complete ontologies; and so on. Therefore, it does not seem meaningful to attempt to introduce a classification of all the approaches.

With MIAKT, Bontcheva [21] presents one of the first implementations of NLG-applications for report generation from existing SW-ontologies—in her case, breast cancer screening and diagnosis report generation. MIAKT takes as input an RDF-description of a patient case introduced by a medical practitioner using an ontology-based user interface, together with a handcrafted medical ontology and a domain lexicon. The medical ontology is a formal description of the breast cancer domain encoded in DAML-OIL, extended by some properties to facilitate the linguistic realization. A reasoner is used to identify and remove duplicated triples in the RDF-input. The remaining triples are grouped and ordered using patterns that rely on properties and their domain classes in the ontology; the adjacent triples with the same property and subject are aggregated. The resulting semantic network is verbalized using an existing surface realizer that operates on conceptual graphs. With the goal to increase the portability of the system, all relations in the input ontologies are mapped to one of four generic and linguistically-motivated relations for which the used surface generator HYLITE+ has in-built support. This approach is extended in ONTOSUM [20], where the text and sentence planning modules can operate on any ontology that contains the same four upper relations.

Argüello et al. [3]'s generator already described in Subsection 3.5 is also situated in the medical domain. It generates clinical narratives from clinical entries specifying the details about a patient's history of present illness in an ophthalmology subdomain. A clinical entry uses a number of mostly already existing OWL-ontologies that model healthcare dialogues and documents as well as clinical terminology. Text planning is performed using a sequence of SPARQL-queries that are executed to select and order the relevant instances. Sentence planning and realization are then performed to transform the results of these queries into sentences using further predefined template-based SPARQL queries which exploit the terminology ontology for lexicalization, although the terms in that ontology might be overridden by the domain and language experts involved in the system's development.

Galanis and Androustsopoulos [55]'s NaturalOWL generates personalized multilingual descriptions of individuals and classes from a linguistically annotated ontology in a number of NLG-applications, including an application for generating descriptions of museum artifacts that builds upon ILEX [100] (see Subection 2.2.1).[15] The different NLG-tasks (e.g., content selection, ordering, lexicalization) are executed in a pipeline and rely heavily on annotations to the input OWL-ontology containing linguistic and task-specific information. For instance, for content selection, the NLG-engineer can make some user modelling annotations for the properties of the ontology, specifying for each property how interesting it is to a given type of user (e.g., child or expert) and how often it can be repeated before it is assimilated. During content selection, the system then selects all the facts about the requested class or individual in the ontology graph up to a set maximum distance, discarding assimilated facts according to its active user profile, ordering the remaining facts according to their interest to the use and selecting the m most interesting facts. Classes and individuals in the OWL ontology are associated with noun phrases, together with their gender and their singular and plural forms. Properties, on the other hand, are assigned one or more template-based micro-plan annotations, which are used in micro-planning to render the lexicalization of facts into an abstract clause specification in a given language. A micro-plan consists of an abstract clause structure containing a verb and information about its inflection and its valency. The valency is used to define the mapping between the semantic arguments of the property and the linguistic arguments of the verb. The latter can be filled with the noun phrases associated to individuals or classes, dynamically generated referring expressions, data-type values or multilingual and personalized canned text. Each micro-plan can be annotated with an appropriateness value for a given type of user. Besides the lexicalization of facts using micro-plans, micro-planning also involves aggregation of resulting clause specifications into complex sentences and generation of referring expressions. A Protegé plug-in is provided such that the annotations NaturalOWL requires on the input ontology can be specified in a graphical environment.

In Bouttaz et al.'s [27] and Bouayad-Agha et al.'s [23, 24] approaches, content selection plays an even more prominent role. Bouttaz et al.'s [27] application is inspired by the work of [63]; see Subsection 4.1. It addresses content selection from an OWL-ontology for the generation of natural language descriptions of digital artifacts and processes in a virtual research environmental (e.g., provenance, date of creation, author) in order to facilitate collaboration and interaction between researchers. The descriptions are generated taking into account: 1) a model of the provenance of the resource described in the ontology itself,[16] 2) model of the user's social context based on an extended version of the FOAF social networking vocabulary,[17] and 3) model of user, project, organization and system policies. The policies are essentially groups of content selection rules based on the provenance and social network models encoded as SPARQL-queries and stored in the metadata repository using TopBraid's SPIN API.[18] They specify access constraints and additional information for a specific user and a given type of artifact. For instance, the principal investigator of a project might want a policy that restricts the view of some information about a process that generated a resource to members of the project, or he might want contact information to be displayed in a resource description if the user is not a member of the project. A user might also want to express her preferences with respect to the display of some information using a policy. When a user requests a textual description of a resource, all the axioms related to that resource are retrieved. A policy manager checks if any of the policies can be activated against the model containing the RDF-graph by running a SPIN-reasoner against the rules associated with the policies, removing or adding information as appropriate.

Bouayad-Agha et al.'s approach [23,24] has been instantiated in two different applications: generation of short football match summaries that vary depending on user's team preference [23] and generation of environmental reports [24]. In both applications, they

[15] Galanis and Androustsopoulos present related work in [1,56].

[16] The model is based on the *Open Provenance Model*, `http://openprovenance.org/`.

[17] `http://www.foaf-project.org/`

[18] `http://spinrdf.org/`

start from an NLG-independent OWL-domain ontology, which is complemented by an extra ontological layer. This additional layer captures, on the one hand, the content that is inferred from the "primitive" concept configurations in the domain ontology, and, on the other hand, *logico-semantic relations* needed for the subsequent generation of coherent discourse (as, e.g., an *implication* relation between a high ozone concentration and a health warning). In the football application, the extended ontology is populated offline by Jena-rules and interpreted using the inference engine provided by Jena[19], while in the environmental application the inferred content is computed on the fly by combining complementary reasoning strategies such as DL and rule-based reasoning and using the Jena API. In the football application, the content selection process is template- and heuristics-based. First, a set of templates is used to bound the KB to the maximal subset of content that is relevant to the match to be reported on. Next, each individual from this preselected subset is evaluated and extracted according to 1) its relation to the user's preferred team, 2) its importance according to a set of empirically determined weights,[20] and 3) its inclusion as argument of a logico-semantic relation. In the environmental application, the relevant raw and additional data are selected or inferred from the ontology to match the user's request and profile. Template-based content selection and text planning (i.e., grouping the ontology individuals into messages, message ordering and mapping of the logico-semantic relations to discourse relations) is performed using SPARQL-queries. In both applications, the resulting plan is extracted from the populated ontology and mapped onto conceptual graphs from which surface generation starts.

In contrast to the above works which operate on existing ontologies, Weal et al. [121] and Dai et al. [35] harvest the information that they incorporate into ontologies before publishing. Weal et al. present a prototypical implementation called ArtEquAKT that generates adaptive biographies of artists from information harvested before from the web. ArtEquAKT uses information extraction technologies to automatically populate an ontology, which is modelled as an extension of the cultural heritage and museum CIDOC-CRM ontology.[21] The extracted facts are associated with their source text fragments. Human authored biography templates are used to reconstruct an artist's biography as a hypertext document combining text fragments in the ontology with sentences generated dynamically from facts also found in the ontology, all according to the user's preferences. The templates are encoded in XML following the *Fundamental Open Hypermedia Model* (FOHM) model for the description of hypermedia documents. They consist of ordered queries to the ontology which retrieve the text fragments associated with the facts being queried, or, if no fragments can be retrieved, dynamically generate a sentence for a given fact using sentence templates. In order to avoid repetition, the overlap of information between text fragments is monitored by a blackboard in which the already mentioned triples are added, such that no new text fragment that contains an already mentioned fact is included. That is, from the NLG point of view, ArtEquAKT reconstructs biographies using templates for text planning and combining the use of text fragments and templates for sentence generation.

Dai et al [35] generate texts from a semantic network, which they call *Semantic Networks Serialization Grammar* (SNLG). Starting from the node of interest in the input semantic network, they iteratively select a semantic star (akin to a star in graph theory) according to a distance function that evaluates to what extent a node is "convincing". The semantic star is "trivialized" into one (or more) patterned semantic trees according to some serialization patterns. The trivialization ensures that each resulting semantic tree can be linguistically realized as sentence, and, therefore, that semantic stars are only selected if they can be trivialized—a strategy that ensures a tight coupling between content determination and linguistic realization. The tree splitting and trivialization is followed by pattern-based aggregation. Both trivialization and aggregation patterns are trained using an annotated corpus. Linguistic realization is in charge of lexicalization (which can be as simple as obtaining the concepts' labels), referring expression generation, which implements [77]'s graph-based approach to this task, and serialization into a language node sequence using the serialization patterns. The framework has been implemented in a system called *NaturalWiki*, which generates texts from fully semanticized wikis in both English and Chinese.

[19] `http://jena.sourceforge.net/`

[20] To determine the weights, supervised learning from a corpus of aligned summaries and tabulated data obtained from the web has been used.

[21] `http://www.cidoc-crm.org/`

The input semantic network is obtained by parsing a corpus of texts on 20 Chinese cities into semantic trees. The trees are merged to avoid repetition and ambiguity using around 500 pattern networks derived (for Chinese) from a different semantically annotated corpus. From the input network, a summary description for each of the 20 cities is generated. It remains to be seen how their approach applies to any RDF-graph, not just one obtained from semantic parsing of texts, and how portable it is to different domains (in case no corpora are available for obtaining the patterns).

Dannélls et al. [40] present a work-in-progress multilingual generation system (so far demonstrated on English and Swedish) that generates verbalizations of responses to queries about museum artifacts from a Museum Reason-able View, which is an assembly of independent datasets and schemas used as a single body of knowledge with respect to reasoning and query evaluation [39].[22] The data in the *Museum Reason-able View* is accessible via a SPARQL endpoint. A query can be formulated by combining predicates from different datasets and ontologies in a single SPARQL-query, retrieving results from all different datasets as a set of triples which is then verbalized. A grammar formalism called the *Grammatical Framework* (GF) is used for NLG. GF comes with a resource library that covers the syntax and lexicons of various languages in order to support the development of new/extended domain-specific grammars and lexicons by non-experts. A semantic grammar specifies discourse patterns, i.e., the order of concepts and their division into sentence units.

Unlike the approaches presented above, Wilcock and Jokinen's [124] and Janzen and Mass's [68] approaches are situated in dialogue applications. Wilcock and Jokinen present an approach to the generation of natural language texts from RDF-XML- and DAML+-OIL-ontologies based on a template-based XSLT pipeline architecture previously developed by Wilcock [123]. For each application domain, a set of XSLT-templates is defined to transform an input XML-structure into an output XML-structure. Each template performs a specific NLG-task (such as referring expression generation, aggregation or discourse structuring). Thus, knowledge of the domain, of NLG-processes, language, and XSLT are tightly coupled. In [124], their application domain is information on public transportation in the Helsinki area; the input to the NLG-component is a time schedule or a list of concepts selected by the dialogue manager for verbalization. As benefit of the use of ontologies, the authors cite the possibility to reason in order to answer complex user queries (as, e.g., *Is the route of the night bus the same as that of the day bus?*, *Are there any buses going from A to B?*, etc.), and to rectify user misconceptions (e.g., by answering *Routes 1 to 10 are tram routes, not bus routes. Do you mean tram number 7?* to the question *When does bus number 7 leave?*).

Janzen and Mass [68] exploit SW-representation, reasoning and querying technologies in a *Conversational Recommendation Agent* (CORA)—a dialogue system that enables shoppers to ask questions about products within an in-store shopping environment. Each product is modelled in OWL-DL as a *Smart Product Description Object* (SPDO); product information is retrieved from the SPDO-pool using SPARQL. Given a corpus of customer-vender conversations, the authors came up with 19 schemas that model the questions that the user can compose and 12 rhetorical plan operators assigned to the schemas to model the answers that satisfy both the buyer's and seller's intentions. From these schemas, the communicative intentions of the customer, which constitute the interface between the NLU-part (schemas) and NLG-part (plan operators) of the system, are also derived. The schemas, plan operators and their intentions are modelled in OWL-DL (and retrieved using SPARQL), drawing upon a domain-independent ontology, the *Semantic QA Structure Model* (SQASM). SQASM also integrates a domain-independent part-of-speech lexicon. The mapping between lexemes and concepts is part of the domain-dependent layer, as are the instances of plan operators, schemas and effects. The user can incrementally compose a question using schema segments. Upon completion, the relevant schema is identified and the appropriate plan operator selected according to the identified intention of the schema. The selected plan operator is expanded and instantiated by queries made to the SPDO-pool. Further information is retrieved via reasoning using standard SW-rules (e.g., SWRL). Which parts of the plan operator are expanded depends on the information that is already in focus on the focus stack that keeps track of the discourse history. An extrinsic evaluation of the quality of the response, the perceived usefulness, etc. of the system has been performed with a group of 16 subjects. The obtained scores were significantly high.

[22] For further references by Dannélls on work related to [40], see [36–38].

4.3. Summary

Table 2 shows a condensed summary of the NLG-applications we discussed in the previous sections according to the NLG-dimensions and features introduced in Section 2. The table is divided between works that use NLG for ontology engineering (Subsection 4.1) in the upper half and works that use NLG for knowledge publishing (Subsection 4.2) in the bottom half. All the features, apart from the input size and input domain independence in Table 1 are considered. Input size and input domain independence are not considered because (nearly) none of the works has a restriction on the size of the input and the input is always domain-dependent. An exception with respect to input size is Sun and Mellish [116], whose input RDF-graph is limited to 10 triples, because their system is a *microplanner* and the target text is a single sentence.

All NLG-approaches to ontology engineering understandably lack *target texts*, i.e., manually crafted texts that the system aims to replicate, and a user profile. Nearly all of them have the common straightforward communicative goal to say all there is to say (see also below). The table also reveals that NLG-approaches to ontology engineering are monolingual (apart from [69]). As pointed out in Section 4.1.1, this is due to their straightforward one-to-one mapping between linguistic objects and ontological knowledge developed using essentially English names and labels; furthermore, the CNLs used in a number of works are all subsets of English.

Unlike NLG-approaches to ontology engineering, many of the approaches to knowledge publishing are developed based on given target texts. Some, like [20, 35,55], are presented as domain-independent, although they are subsequently tested in specific domains with specific target texts such as encyclopedic descriptions [35] or museum artifact descriptions [55].

Most approaches start from an input representation (i.e., ontology) which has not been designed deliberately for NLG. A number of works, however, impose restrictions on the naming of labels in the ontology with the aim to ensure the grammaticality of the generated text; see [62,64,103]. Others extend a task-independent ontology by task-specific and/or linguistic knowledge. This additional knowledge can consist of annotated user or linguistic models [55,103], a linguistically motivated set of generic concepts and/or relations [21,47], additional domain-specific concepts and relations whose instances can be inferred from the domain knowledge and are communicated in target texts [23,24], etc.

The target audience of NLG-approaches to knowledge publishing is constituted by end users—lay persons or domain experts (e.g., medical practitioners [3], social science researchers [27], environmental experts [24]), whereas NLG-approaches to ontology engineering target domain experts that act mainly as ontology developers or imaginary lay persons.

Some approaches are concerned with the verbalization of the entire input graph [20,21,48,116] (minus redundancies in the case of [20,21]), thus excluding content selection from the scope of their work. Most are interested in the description of a specific node in the ontology graph, be it a class or a specific instance. Indeed, all approaches but [68] are data driven. Their communicative goal is one of the following three: (i) to say all there is to say about an entity or graph (as, e.g., [62], [115]); (ii) to verbalize the most relevant facts (as, e.g., [55], [35]); or (iii) to verbalize the most typical facts according to some target texts (as, e.g., [121], [40]). While (iii) is typically achieved by closed planning using straightforward templates, (ii) can be achieved either by open planning using the input graph topology with or without pondered nodes based on the nodes' inherent value and/or a user model [35,55], or by closed planning with restrictions on the applications of each template [24,27]; for (i), obviously, no content selection is needed, although discourse structuring by way of simple ordering rules is in order. Some approaches such as [23,24,27] combine the requirement to communicate the most typical facts in a target text (e.g., result and team names for football match summaries) with the requirement to communicate the most relevant facts.

When used, the type of user profile is very heterogeneous and can be a list of properties that are to be filtered out [20], discourse history [68], user perspective [23,26], interest [55] or social context [27].[23]

As far as system's output is concerned, fluency tends to be higher in systems that use lexicalization, aggregation and linguistic realization together with lexical and grammatical resources. However, our assessment of the level of fluency relies not only on our knowledge

[23]User profile refers to NLG-systems that generate different texts for different users given the same input using an explicit user profile/model, thus excluding works like Argüello [3], whose input data is already specific to a user, or Jarrar et al. [69], who theoretically consider taking into account different users or scenarios by defining specific templates for each of them.

Approach	Input		Context					Output			
	Type	TI	T. Texts	T. Aud.	Verb. Req.	Com.Goal	UP	Flu.	Size	Coh.	Lang.
Hewlett [62]	OWL-DL ont.	Yes	No	LP, DE	Class description	Say all	No	+	P	Yes	Eng
Jarrar [69]	ORM	Yes	No	DE	Constraint	N/A	No	– –	S	N/A	Multi.
Sun [116]	RDF Graph	Yes	No	LP,DE	Graph	Say all	No	+	S	N/A	Eng
Kaljurand [71]	OWL-DL ont.	Yes	No	DE	Class description	Say all	No	–	P	Yes	Eng
Ang [2]	OWL-DL ont.	Yes	No	DE	Queries of a term	Say all	No	+	P	N/A	Eng
Mellish [89,90]	OWL-DL ont.	Yes	No	LP,DE	Class subsumers	Say all	No	N/A	N/A	N/A	N/A
Davis [42]	OWL-DL ont.	Yes	No	DE	Classes, Properties Instance	Say all	No	–	P	No	Eng
Stevens [115]	(Subset of) OWL-DL ont.	Yes	No	DE	Class description	Say all	No	–	P	Yes	Eng
Power [103]	(restricted) OWL-DL graph	Mixed	No	DE	Class axiom Individual assertion	N/A	No	–	S	N/A	Eng
Hielkema [63,64]	OWL-Lite/RDF graph	No	No	DE	Metadata description about intellectual artifact	Say all	No	–	T	Yes	Eng
Dongilli [48]	OWL/RDF graph	Yes	No	DE	Query	Say all	No	+	S	N/A	Eng
Wilcock [124]	RDF/XML graph, DAML-OIL ont.	Yes	No	LP	Answer to user question about individual or class	Say all about individual, Summarize class	No	+	P	Yes	Eng
Bontcheva [21]	DAML-OIL/RDF graph	Yes	Clinical case reports	DE	The graph	Say all (non redundant)	No	+	P	Yes	Eng
Bontcheva [20]	OWL/RDF graph	Yes	No	LP	Graph	Say all (non redundant)	Yes	+	T	Yes	Eng
Argüello [3]	OWL/RDF graph	No	Clinical narratives	DE	Description of current illness	Say typical facts	No	+	P	Yes	Eng
Galanis [55]	OWL-DL/RDF ont.	Mixed	No	LP,DE	Specific entity	Say most relevant facts	Yes	+	T	Yes	Eng, Gre
Bouttaz [27]	OWL/RDF graph	Yes	No	DE	Metadata description about digital artifact	Say typical, most relevant facts	Yes	N/A	N/A	N/A	N/A
Bouayad-Agha [23]	OWL-DL ont.	Mixed	Football summaries	LP	Match summary	Say typical, most relevant facts	Yes	+	P	Yes	Spa
Bouayad-Agha [24]	OWL-DL ont.	Mixed	Environmental bulletins	LP,DE	Environmental info given date and location	Say typical, most relevant facts	Yes	+	T	Yes	Eng, Fin, Swe
Weal [121]	Protegé frames RDF ont.	Yes	Artist bios (summary, chronology)	LP	Artist name	Say typical facts	Yes	+	T	Yes	Eng
Janzen [68]	OWL-DL ont.	Yes	Customer-Vendor dialogues	LP	Answer to customer query	Maximize both parties' intentions	Yes	+	P	Yes	Eng
Dai [35]	Semantic network	Yes	No	LP	Specific entity in network	Say most relevant facts	No	+	P	Yes	Chn, Eng
Dannélls [40]	OWL/RDF ontology	Yes	Description of museum artifacts	LP	Museum artifact description	Say typical facts	Yes	+	P	Yes	Eng, Swe

Table 2: Summary of Semantic Web NLG-Applications (Section 4) wrt NLG-dimensions and features in Table 1
Columns: Approach (by first referenced author), Input Type, Task Independence (TI), Target Texts (T.Texts), Target Audience (T. Aud.)={LP=Lay Person, DE=Domain Expert}, Verbalization Request (Verb. Req), Communicative Goal (Com. goal), User Profile (UP), Fluency (Flu.)=High(+),Medium(–),Low(– –), Output Size={S(entence), P(aragraph), T(ext)}, Coherence (Coh.), Language (Lang.) = {Eng=English, Multi=Multilingual, Gre=Greek, Spa=Spanish, Fin=Finnish, Swe=Swedish, Chn=Chinese}

of the performed NLG-tasks and their scope, but also on the sample texts provided in the reference articles.[24] For instance, Davis et al. [42] state that their ontology verbalizer uses XML-templates that perform aggregation, impose coreference, and do basic morphological inflection. However, the example output they provide shows that class names are treated as proper names (e.g., 'Researcher' *attends* 'Conference'), which naturally limits the fluency of the output text (as, would be the case in 'SeniorResearcher' *attends* 'Conference'). Similarly, in [63], aggregation is limited to cases where the source and property label are the same, thus limiting the possibilities (as, e.g., for the following two statements: *It was deposited on 9 March 2007*. *It was deposited by John Farrington*). Other non-strictly NLG-tasks can contribute to improve fluency, such as final spell checking [2], removal or monitoring of repetitions/redundancies [20,21,62,121] or presentation of sentence coordinated constituents in a bulleted list [62].

The size of the output of the NLG-system can either be a sentence (possibly a complex one), a paragraph (i.e., a thematically related sequence of sentences), or a text (i.e., a sequence of paragraphs). Coherence only applies above the sentence level, even though for complex sentences, some syntactically enforced coherence between the clauses of a sentence is in order. As a matter of fact, coherence is often attained using topic/triple ordering templates (see, e.g., [21,40]). But it can also be achieved using simple ordering rules (as, e.g., class first, then properties) for prototypical texts (e.g., [115,124]), partial order constraints as annotations on the ontology [47,55], or more complex graph-based search (as, e.g., [35]). In some of these works, coherence is further leveraged with aggregation and referring expression generation (as, e.g., in [35,55]).

5. Burning issues

The previous sections have shown that NLG is increasingly using SW-technologies. However, a number of issues need to be addressed before we can speak of a real symbiosis between NLG and SW. These "burning issues" concern, on the one hand, a tighter interrelation between NLG and SW-representations, and, on the other hand, a more intense integration of NLG into SW-applications. Let us discuss a few these isses in some detail.

5.1. Codification of NLG-relevant knowledge in SW

In Subsection 3.1, we pointed out that the possibility of a uniform codification of NLG-relevant information in terms of a SW-representation makes a significant difference compared to traditional representations. Wilks and Brewster [125] argue that it is NLP, which is the foundation of the SW. This may lead us to the conclusion that SW-resources contain a significant share of linguistic knowledge that is also relevant to NLG. However, as a matter of fact, there has been only limited work on the systematic codification of linguistic/NLG-related information in terms of OWL ontologies. To be mentioned are, first of all, the initiatives on the representation of lexico-semantic information—starting with Bateman's *Generalized Upper Model* [10] [25] already discussed in Subsection 3.1, whose history goes back to the mid-nineties, and continuing with LexOnto [32], LexInfo [28],[26], LIR [95], etc.; see also Subsection 3.1. As our discussion in 3.1 shows, the codification of other types of NLG-relevant information has so far been approach-specific and selective in the sense that, to the best of our knowledge, none of the existing works codifies all relevant information (such as discourse relation models, user models, planning operators, context models, etc.) in a SW-representation. This reduces the efficiency of the generators and hampers the exchange and/or plug-in of individual modules. A concerted action is thus needed to determine which types of NLG-relevant information should be captured in a SW-representation and to agree on the codification.

5.2. Standardization of NLG-interfaces using SW

Immediately related to the central question of standardization of the codification of NLG-relevant information is the question of the interfaces of individual modules in NLG. In the past, efforts such as RAGS ("Reference Architecture for Generation Systems") [91] have been made to come up with a formal specification of consensus representations (conceptual, rhetorical, document, semantic, syntactic and quote representations) that can serve as input or output

[24]This means that our interpretation of a system's output as being of low to medium fluency is more justified than as being of high fluency.

[25]`http://www.ontospace.uni-bremen.de/ontology/stable/GUM-3.owl`

[26]`http://lexinfo.net`

of the different NLG-modules, thus allowing the interchange and communication between modules that implement the RAGS framework. However, the RAGS-representation models have not been taken up by the NLG-community. According to one of the main authors of RAGS C. Mellish [88], this is due to the complexity of the RAGS-framework, lack of tools to support its implementation (i.e., APIs in different programming languages, consistency checking, query engines, etc.), its idiosyncratic use of XML and its inability to define how to bring together RAGS-representations with non-RAGS ones. Mellish suggests that these difficulties can be remedied by recasting quite naturally RAGS-data in terms of RDF and formalizing the RAGS-representations using SW-ontologies. As a matter of fact, this would result in consensus representations and models of linguistic resources and input/output of NLG-modules or -tasks interfaces, and increase the performance of the generators. Without such a recasting it will be very hard to achieve a breakthrough of NLG-technologies.

5.3. NLG and linked open data and texts

Virtually all efforts in NLG from SW-datasets are still restricted to isolated datasets, leaving much ground to cover before mature technologies are available for the production of text from Linked Data. When NLG taps the full potential of linked data, it will have to adapt to vast amounts of data described using multiple vocabularies and encoded in knowledge representations of varying degrees of expressivity (e.g., SKOS, RDFS, and different OWL dialects). This will have an impact not only on the selection of contents, but also on other NLG-tasks which operate or reason on the same input representation—for instance, ordering contents for their inclusion in the text or determining their lexical realization. In particular, generating from linked data will require that implementations of these tasks are aware of the linkage between vocabularies in order to treat information from heterogeneous data sources as semantically equivalent or related. Thus, if a vocabulary link states that two classes from two different vocabularies used in distinct datasets are equivalent, reasoners and query engines used in NLG must be able to consider all instances of both classes as individuals of the same class. This will be necessary, for instance, when selecting or ordering facts according to the classes of the individuals referenced in the fact.

Another prominent feature of a number of open linked datasets is that their data are related to Hypertext Web documents. For instance, in DBPedia RDF-triples are paired with the Wikipedia articles they were extracted from. The text found in the paired documents can be seen as a verbalization of the data, and thus as training material for empirical NLG: documents can provide an insight on which data is most relevant, how it is ordered, and which are the linguistic expressions that are used to communicate the data. Some works in NLG outside the scope of the SW have explored the creation and use of text corpora aligned with data for some specific NLG-tasks (as, e.g., [5] for the selection of contents from databases). The methods for the alignment of texts with the data communicated in them and for the extraction of useful heuristics could also be applied in the context of Linked Data and Web documents.

The recently announced content selection challenge from SW-data paired with corresponding data sets [22] is expected to advance the state-of-the-art in this field and bring the NLG- and SW-communities closer to each other.

5.4. Summarization of large volumes of data

In Section 3.3, we argued that SW offers the chance and, at the same time, poses the challenge to scale up NLG-techniques to large volumes of data. Conversely, there is a growing need in the SW-community for technologies that give humans easy access to the machine-oriented Web of data [61]. NLG provides the means for presenting semantic data in an organized, coherent and accessible way thanks to its capabilities of rendering content as multilingual or even multimodal information. We have already shown that NLG-technologies have been successfully applied to the generation of information from large data sets(see Subsection 4.2). This success is also in part due to the strategies for content selection developed in NLG over the years (see Subsection 2.2.1). By adapting these strategies to Semantic Web data, NLG can be used to generate summaries that communicate the most important content in a dataset while filtering out the less relevant content. An illustration of how this can be done by applying reasoning techniques is given, e.g., in [23,24]. On the other hand, works on ontology summarization can also contribute to the generation of summaries from large datasets. While strategies for content selection in NLG take as criteria what content is communicated in target texts, approaches to ontology summarization employ topographical measures over RDF-graphs in order to assess what content is most important; see, e.g. [31,128]. Ontology summarization could be used to leverage NLG and improve the suitability of text-based summaries as an accessible way of presenting data to humans.

5.5. Combining content distillation and generation

The vocabulary and data in the SW are often obtained automatically or semi-automatically from textual information in the Hyperdocument Web using Knowledge Acquisition (KA), Information Extraction (IE), or Text Mining (TM) techniques [125]. Even when SW-data are encoded manually, they often originate from the analysis or manual annotation of existing Web documents. In contrast, NLG starts from data and produces text documents. This situation has been referred to as the *language loop* [19]. NLG plays an increasingly central role in the language loop because many original web texts require a paraphrase, summarization or translation in order to serve the needs of the targeted user. The SW acts as a sort of *interlingua*, which helps in bridging the gap between the source and destination texts. NLG is thus to be seen here as regeneration guided by SW-data, where NLG-techniques are combined with IE- and TM-techniques in order to produce new textual material. This material may contain both dynamically generated text and text fragments obtained from original Web documents. To the best of our knowledge, the ArtEquAKT system presented in Subsection 4.2 is the only NLG-system to explore this increasingly prominent topic.

5.6. Adapting to context

Tailoring the content and wording to the preferences and particularities of end users as well as to other contextual conditions is instrumental for personalized NLG. NLG has a long tradition of working on experimental techniques that adapt the output to the context by varying the information communicated in the text, its order and general organization, the language, the general writing style and lexical choices, and even the mode (e.g. text, images). Many of the SW-oriented NLG-proposals discussed in Subsection 4.2 are also, to a certain extent, context-sensitive. The contextual information they deal with includes the target language (as, e.g. [24]), user preferences and partial order restrictions for certain types of contents (as, e.g. [55]), or a history of the content communicated to a specific user (as, e.g. [36]). This information is used to influence common text planning tasks such as content selection or content ordering or to control the use of certain HTML-layout features (as in ONTOSUM [20]). However, the degree of their context-orientation is, in general, considerably lower than in traditional (admittedly much more experimental) implementations.

On the other hand, the SW constitutes an excellent means for the codification of detailed user-oriented contexts. Furthermore, both existing vocabularies and the input data sets can be seamlessly integrated using the same mechanisms used to publish Linked Data, i.e. URIs and identity and vocabulary links. This is illustrated in [27], where complex contextual knowledge for content selection (i.e. policies based on provenance information) is encoded using SW-technologies and in part described using existing vocabularies such as FOAF. We are convinced that SW-oriented NLG will be required to work further in this direction.

6. Conclusions

One of the original and ultimate goals of the SW was to allow agents to carry out sophisticated tasks for users based on retrieving and processing meaningful content, and communicating the results in a user-friendly way [15]. Some applications of this kind have been developed in limited domains and for limited tasks; see, for instance [127]'s system that answers questions about pop trivia, [68]'s system that answers questions about products for a virtual in-store shopping environment, or [121]'s ArtEquAkt-system that combines SW-knowledge and text fragments to generate adaptive artist biographies. The NLG-component used in these applications is developed from scratch and relies on relatively simple templates. We are still a long way from embodied characters that engage in conversation in a multimodal way to present informative content tailored to the context [33]. However this vision might be attainable if the Semantic Web and Natural Language Generation community join their efforts to develop robust data-driven approaches, and standardized representations of input (and also certain intermediate) structures, context, and processes.

References

[1] ANDROUTSOPOULOS, I., KALLONIS, S., AND KARKALETSIS, V. Exploiting OWL Ontologies in the Multilingual Gen-

eration of Object Descriptions. In *Proceedings of the 10th European Workshop on Natural Language Generation* (2005), pp. 150–155.

[2] ANG, W., KANAGASABAI, R., AND BAKER, C. Knowledge translation: Computing the query potential of bio-ontologies. In *Proceedings of the Workshop on Semantic Web Applications and Tools for Life Sciences* (2008).

[3] ARGÜELLO, M. F., DES, J., PRIETO, M. J. F., PEREZ, R., AND LEKKAS., S. An Ontology-Based Approach to Natural Language Generation from Coded Data in Electronic Health Records. In *Proceedings of the UKSim 5th European Symposium on Computer Modeling and Simulation* (2011), pp. 366–371.

[4] BAO, J., SMART, P., BRAINES, D., AND SHADBOLT, N. A Controlled Natural Language Interface for Semantic Media Wiki Using the Rabbit Language. In *Workshop on Controlled Natural Language (CNL '09)* (June 2009).

[5] BARZILAY, R., AND LAPATA, M. Collective content selection for concept-to-text generation. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing* (Vancouver, 2005).

[6] BASILE, V., AND BOS, J. Towards generating text from discourse representation structures. In *Proceedings of the 13th European Workshop on Natural Language Generation* (Nancy, France, 2011).

[7] BATEMAN, J. Upper Modelling: Organizing Knowledge for Natural Language Processing. In *Proceedings of the 5th International Workshop on Natural Language Generation, 3-6 June 1990* (Dawson, PA., 1990).

[8] BATEMAN, J. Uncovering Textual Meanings: A Case Study Involving Systemic-Functional Resources for the Generation of Japanese Texts. In *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, C. Paris, W. Swartout, and W. Mann, Eds. Kluwer Academic Publishers, 1991.

[9] BATEMAN, J. Enabling technology for multilingual natural language generation: The KPML development environment. *Natural Language Engineering 3*, 1 (1997), 15–55.

[10] BATEMAN, J. A., MAGNINI, B., AND FABRIS, G. The Generalized Upper Model Knowledge Base: Organization and Use. In *Towards Very Large Knowledge Bases*, N. Mars, Ed. IOS Press, Amsterdam, 1995, pp. 60–72.

[11] BELZ, A., BOHNET, B., MILLE, S., WANNER, L., AND WHITE, M. The Surface Realization Task: Recent Developments and Future Plans. In *Proceedings of the 7th International Conference on Natural Language Generation* (Starved Rock, IL, 2012).

[12] BELZ, A., AND REITER, E. Comparing automatic and human evaluation in NLG. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)* (2006), pp. 60–72.

[13] BELZ, A., AND REITER, E. Comparing Automatic and Human Evaluation in NLG. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistic* (2006), EACL'06, pp. 313–320.

[14] BELZ, A., WHITE, M., ESPINOSA, D., KOW, E., HOGAN, D., AND STENT, A. The First Surface Realization Shared Task: Overview and Evaluation Results. In *Proceedings of the 13th International Workshop on Natural Language Generation* (Nancy, France, 2011).

[15] BERNERS-LEE, T., HENDLER, J., AND LASSILA, O. The Semantic Web. *Scientific American* (2001), 29–37.

[16] BERNSTEIN, A., KAUFMANN, E., AND KAISER, C. Querying the semantic web with ginseng: A guided input natural language search engine. In *15th Workshop on Information Technologies and Systems, Las Vegas, NV* (2005), pp. 112–126.

[17] BILIDAS, D., THEOLOGOU, M., AND KARKALETSIS, V. Enriching OWL Ontologies with Linguistic and User-Related Annotations: The ELEON System. In *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on* (Oct. 2007), vol. 2, pp. 464 –467.

[18] BIZER, C., HEATH, T., AND BERNERS-LEE, T. Linked Data—The Story so Far. *Interational Journal of Semantic Web for Information Systems 5*, 3 (2009), 1–22.

[19] BONTCHEVA, D. K., AND CUNNINGHAM, H. The semantic web: A new opportunity and challenge for human language technology. In *Proceedings of the Third International Semantic Web Conference (ISWC)* (2003).

[20] BONTCHEVA, K. Generating tailored textual summaries from ontologies. *The Semantic Web: Research and Applications* (2005), 531–545.

[21] BONTCHEVA, K., AND WILKS, Y. Automatic report generation from ontologies: The MIAKT approach. In *Proceedings of the Nineth International Conference on Applications of Natural Language to Information Systems (NLDB'2004). Lecture Notes in Computer Science 3136* (Manchester, UK, 2004).

[22] BOUAYAD-AGHA, N., CASAMAYOR, G., MELLISH, C., AND WANNER, L. Content Selection from Semantic Web Data. In *Special Track on Future Generation Challenges Proposals, Proceedings of the International Natural Language Generation Conference* (2012).

[23] BOUAYAD-AGHA, N., CASAMAYOR, G., MILLE, S., AND WANNER, L. Perspective-oriented generation of football match summaries: Old tasks, new challenges. *ACM Transactions on Speech and Language Processing (TSLP) 9*, 2 (2012).

[24] BOUAYAD-AGHA, N., CASAMAYOR, G., ROSPOCHER, M., SAGGION, H., SERAFINI, L., AND WANNER, L. From ontology to NL: Generation of multilingual user-oriented environmental reports. In *Proceedings of the 17th International conference on Applications of Natural Language Processing to Information Systems (NLDB'2012). Lecture Notes in Computer Science series, vol. 7337* (Groningen, Netherlands, 2012).

[25] BOUAYAD-AGHA, N., CASAMAYOR, G., AND WANNER, L. Content determination from an ontology-based knowledge base for the generation of football summaries. In *Proceedings of the 13th European Natural Language Generation Workshop (ENLG)* (Nancy, France, 2011).

[26] BOUAYAD-AGHA, N., CASAMAYOR, G., WANNER, L., DÍEZ, F., AND HERNÁNDEZ, S. FootbOWL: Using a generic ontology of football competition for planning match summaries. In *Proceedings of the Extended Semantic Web Conference(ESWC)* (Heraklion, Crete, Greece, 2011).

[27] BOUTTAZ, T., PIGNOTTI, E., MELLISH, C., AND EDWARDS, P. A policy-based approach to context dependent natural language generation. In *Proceedings of the 13th European Workshop on Natural Language Generation (ENLG 2011)* (Nancy, France, 2011), pp. 151–157.

[28] BUITELAAR, P., CIMIANO, P., HAASE, P., AND SINTEK, M. Towards linguistically grounded ontologies. In *Proceed-*

ings of the 6th European Semantic Web Conference (ESWC) on The Semantic Web: Research and Applications (Heraklion, Crete, Greece, 2009).

[29] BUITELAAR, P., SINTEK, M., AND KIESEL, M. A lexicon model for multilingual/multimedia ontologies. In *Proceedings of the 3rd European Semantic Web Conference (ESWC06)* (Budva, Montenegro, 2006).

[30] BUNT, H. Utterance generation from semantic representations augmented with pragmatic information. In *Natural Language Generation*, G. Kempen, Ed. Martinus Nijhoff Publishers, 1987.

[31] CHENG, G., GE, W., AND QU, Y. Generating summaries for ontology search. In *Proceedings of the 20th international conference companion on World wide web* (New York, NY, USA, 2011), WWW '11, ACM, pp. 27–28.

[32] CIMIANO, P., HAASE, P., HEROLD, M., MANTEL, M., AND BUITELAAR, P. Lexonto: A Model for Ontology Lexicons for Ontology-Based NLP. In *Proceedings of the OntoLex (From Text to Knowledge: The Lexicon/Ontology Interface) Workshop at the International Semantic Web Conference (ISWC)* (2007).

[33] CIMIANO, P., AND KOPP, S. Accessing the web of data through embodied virtual characters. *Semantic Web 1*, 1 (2010), 83–88.

[34] CLAIRE GARDENT, E. B., AND PEREZ-BELTRACHINI, L. Natural Language Generation and Natural Language Interfaces to Knowledge Bases, 2011.

[35] DAI, Y., ZHANG, S., CHEN, J., CHEN, T., AND ZHANG, W. Semantic Network Language Generation Based on a Semantic Networks Serialization Grammar. *World Wide Web 13*, 3 (2010), 307–341.

[36] DANNÉLLS, D. The Value of Weights in Automatically Generated Text Structures. *Computational Linguistics and Intelligent Text Processing* (2009), 233–244.

[37] DANNÉLLS, D. Applying semantic frame theory to automate natural language template generation from ontology statements. In *Proceedings of the 6th International Natural Language Generation Conference* (Stroudsburg, PA, USA, 2010), INLG '10, Association for Computational Linguistics, pp. 179–183.

[38] DANNÉLLS, D. On generating coherent multilingual descriptions of museum objects from semantic web ontologies. In *INLG 2012 Proceedings of the Seventh International Natural Language Generation Conference* (Utica, IL, May 2012), Association for Computational Linguistics, pp. 76–84.

[39] DANNÉLLS, D., AND DAMOVA, M. Reason-Able View of Linked Data for Cultural Heritage. In *Proceedings of the Third International Conference on Software, Services and Semantic Technologies (S3T)* (2011).

[40] DANNÉLLS, D., DAMOVA, M., ENACHE, R., AND CHECHEV, M. Multilingual Online Generation from Semantic Web Ontologies. In *Proceedings of the 21st international conference companion on World Wide Web* (New York, NY, USA, 2012), WWW '12 Companion, ACM, pp. 239–242.

[41] DAVIES, J., DUKE, A., KINGS, N., MLADENIC, D., BONTCHEVA, K., GRCAR, M., BENJAMINS, R., CONTRERAS, J., CIVICO, M., AND GLOVER, T. Next generation knowledge access. *Journal of Knowledge Management 9*, 5 (Oct. 2005), 64–84.

[42] DAVIS, B., IQBAL, A., FUNK, A., TABLAN, V., BONTCHEVA, K., CUNNINGHAM, H., AND HANDSCHUH, S. RoundTrip ontology authoring. In *Proceedings of the International Semantic Web Conference (ISWC)*, vol. 5318 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2008, pp. 50–65.

[43] DE COI, J., FUCHS, N., KALJURAND, K., AND KUHN, T. Controlled English for reasoning on the Semantic Web. In *Semantic techniques for the web, the REWERSE Perspective*, F. Bry and J. Maluszynski, Eds. Springer, 2009, pp. 276–308.

[44] DEEMTER, K. V., KRAHMER, E., AND THEUNE, M. Real vs. template-based NLG: A false opposition? *Computational Linguistics 31*, 1 (2005).

[45] DEMIR, S., CARBERRY, S., AND MCCOY, K. A discourse-aware graph-based content-selection framework. In *Proceedings of the 6th International Natural Language Generation Conference (INLG)* (Trim, Co. Meath, Ireland, 2010), pp. 17–27.

[46] DENAUX, R., DIMITROVA, V., COHN, A., DOLBEAR, C., AND HART, G. Rabbit to OWL: Ontology Authoring with a CNL-Based Tool. In *Workshop on Controlled Natural Language (CNL)* (Marettimo Island, Italy, 2009), pp. 246–264.

[47] DONGILLI, P. Natural language rendering of a conjunctive query. *KRDB Research Centre Technical Report No. KRDB08-3). Bozen, IT: Free University of Bozen-Bolzano* (2008).

[48] DONGILLI, P., AND FRANCONI, E. An Intelligent Query Interface with Natural Language Support. In *Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference* (Florida, USA, 2006), FLAIRS'06.

[49] ELHADAD, M., MCKEOWN, K., AND ROBIN, J. Floating Constraints in Lexical Choice. *Computational Linguistics 23*, 2 (1997), 195–239.

[50] FRANCONI, E., GUAGLIARDO, P., AND TREVISAN, M. An intelligent query interface based on ontology navigation. In *Workshop on Visual Interfaces to the Social and Semantic Web, VISSW* (2010), vol. 10.

[51] FRANCONI, E., GUAGLIARDO, P., AND TREVISAN, M. Quelo: a NL-based intelligent query interface. In *Proceedings of the Second Workshop on Controlled Natural Languages* (2010).

[52] FRANCOPOULO, G., BEL, N., GEORGE, M., CALZOLARI, N., MONACHINI, M., PET, M., AND SORIA, C. Lexical markup framework: ISO standard for semantic information in NLP lexicons. In *Proceedings of the Workshop of the GLDV Working Group on Lexicography at the Biennial Spring Conference of the GLDV* (2007).

[53] FROHLICH, M., AND REIND VAN DE, R. Using Multiple Ontologies in a Framework for Natural Language Generation. Tech. rep., Computer Science Department, Stanford University, 1998.

[54] FUNK, A., TABLAN, V., BONTCHEVA, K., CUNNINGHAM, H., DAVIS, B., AND HANDSCHUH, S. Clone: Controlled Language for Ontology Editing. In *Proceedings of the 6th International Semantic Web Conference (ISWC) and the 2nd Asian Semantic Web Conference (ASWC)* (Busan, Korea, 2007), pp. 142–155.

[55] GALANIS, D., AND ANDROUTSOPOULOS, I. Generating multilingual personalized descriptions from OWL ontologies on the semantic web: The natural OWL system, 2007.

[56] GALANIS, D., KARAKATSIOTIS, G., LAMPOURAS, G., AND ANDROUTSOPOULOS, I. An open-source natural

language generator for OWL ontologies and its use in Protégé and Second Life. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations Session* (Stroudsburg, PA, USA, 2009), EACL '09, Association for Computational Linguistics, pp. 17–20.

[57] GOLDMAN, N. Conceptual Generation. In *Conceptual Information Processing*, R. Schank, Ed. North-Holland Publishing Co, Amsterdam, 1975.

[58] HALLETT, C., SCOTT, D., AND POWER, R. Intuitive querying of e-Health data repositories. In *Proceedings of the UK E-Science All-Hands Meeting* (Notthingham,UK, 2005).

[59] HALLETT, C., SCOTT, D., AND POWER, R. Composing queries through conceptual authoring. *Computational Linguistics 33*, 1 (2007), 105–133.

[60] HART, G., JOHNSON, M., AND DOLBEAR, C. Rabbit: Developing a Control Natural Language for Authoring Ontologies. In *Proceedings of the 5th European Semantic Web Conference (ESWC)* (Tenerife, Canary Islands, 2008), pp. 348–360.

[61] HEATH, T., DOMINGUE, J., AND SHABAJEE, P. User interaction and uptake challenges to successfully deploying Semantic Web technologies. In *SWUI 2006, the 3rd International Semantic Web User Interaction Workshop (colocated with ISWC2006)* (2006).

[62] HEWLETT, D., KALYANPUR, A., KOLOVSKI, V., AND HALASCHEK-WIENER, C. Effective nl paraphrasing of ontologies on the Semantic Web. In *Workshop on End-User Semantic Web Interaction, 4th Int. Semantic Web conference, Galway, Ireland* (2005).

[63] HIELKEMA, F., MELLISH, C., AND EDWARDS, P. Using WYSIWYM to create an open-ended interface for the semantic grid. In *Proceedings of the Eleventh European Workshop on Natural Language Generation* (Stroudsburg, PA, USA, 2007), ENLG '07, Association for Computational Linguistics, pp. 69–72.

[64] HIELKEMA, F., MELLISH, C., AND EDWARDS, P. Evaluating an ontology-driven WYSIWYM interface. In *Proceedings of the Fifth International Natural Language Generation Conference* (2008), INLG '08, pp. 138–146.

[65] HORACEK, H. The architecture of a generation component in a complete natural language dialogue system. In *Current Research in Natural Language Generation*, R. Dale, C. Mellish, and M. Zock, Eds. Academic Press, 1990, pp. 193–227.

[66] HOVY, E. *Generating Natural Language under Pragmatic Constraints*. Lawrence Erlbaum, Hillsdale, New Jersey, 1988.

[67] HOVY, E. Approaches to the Planning of Coherent Text. In *Natural Language Generation in Artificial Intelligence and Computational Lingiustics*, C. Paris, W. Swartout, and W. Mann, Eds. Kluwer Academic Publishers, Dordrecht, 1991, pp. 83–102.

[68] JANZEN, S., AND MAASS, W. Ontology-Based Natural Language Processing for In-store Shopping Situations. In *Semantic Computing, 2009. ICSC '09. IEEE International Conference on* (Sept. 2009), pp. 361–366.

[69] JARRAR, M., MARIA, C., AND DONGILLI, K. Multilingual verbalization of ORM conceptual models and axiomatized ontologies, 2006.

[70] JOSHI, A., AND SCHABES, Y. Tree-Adjoining Grammars. In *Handbook of Formal Languages*, G. Rozenberg and A. Salomaa, Eds. Springer, Berlin, 1997, pp. 69–124.

[71] KALJURAND, K., AND FUCHS, N. E. Verbalizing OWL in Attempto Controlled English. In *Proceedings of Third International Workshop on OWL: Experiences and Directions, Innsbruck, Austria* (2007).

[72] KASPER, R. A Flexible Interface for Linking Applications to PENMAN's Sentence Generator. In *Proceedings of the DARPA Workshop on Speech and Natural Language* (1989).

[73] KAUFMANN, E., AND BERNSTEIN, A. Evaluating the usability of natural language query languages and interfaces to Semantic Web knowledge bases. *Web Semantics 8*, 4 (Nov. 2010), 377–393.

[74] KAUFMANN, E., BERNSTEIN, A., AND ZUMSTEIN, R. Querix: A natural language interface to query ontologies based on clarification dialogs. In *5th International Semantic Web Conference (ISWC 2006)* (2006), pp. 980–981.

[75] KITTREDGE, R., KORELSKY, T., AND RAMBOW, O. On the need for domain communication knowledge. *Computational Intelligence 7*, 4 (1991), 305 – 314.

[76] KOSSEIM, L., AND LAPALME, G. Choosing rhetorical structures to plan instructional texts. *Computational Intelligence 16*, 3 (2000), 408–445.

[77] KRAHMER, E., ERK, S., AND VERLEG, A. Graph-Based Generation of Referring Expressions. *Computational Linguistics 29* (2003), 53–72.

[78] KUHN, T. Acewiki: A Natural and Expressive Semantic Wiki. In *Proceedings of Semantic Web User Interaction at CHI 2008: Exploring HCI Challenges* (2008).

[79] LAREAU, F., AND WANNER, L. Towards a Generic Multilingual Dependency Grammar for Text Generation. In *Grammatical Engineering Across Frameworks (GEAF '07)*, T. H. King and E. Bender, Eds. CSLI Publications in Computational Linguistics, Stanford, 2007.

[80] LOPEZ, V., UREN, V., SABOU, M., AND MOTTA, E. Is Question Answering Fit for the Semantic Web?: A Survey. *Semantic Web 2*, 2 (2011), 125–155.

[81] MACGREGOR, R. Retrospective on Loom. Tech. rep., Information Sciences Institute (ISI), University of Southern California, Los Angeles, 1999.

[82] MANN, W., AND THOMPSON, S. Rhetorical Structure Theory: A theory of text organization. In *The Structure of Discourse*, L. Polanyi, Ed. Ablex Publishing Corporation, Norwood, New Jersey, 1987.

[83] MARCU, D. From local to global coherence: A bottom-up approach to text planning. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Innovative Applications of Artificial Intelligence Conference (AAAI/IAAI'97)* (Providence, Rhode Island, 1997), pp. 629–635.

[84] MATTHIESSEN, C., AND BATEMAN, J. *Text Generation and Systemic-Functional Linguistics: experiences from English and Japanese*. Frances Pinter Publishers and St. Martin's Press, London and New York, 1991.

[85] MAYBURY, M. Communicative acts for explanation generation. *International Journal of Man-Machine Studies 37*, 2 (1992), 135–172.

[86] MCCRAE, J., SPOHR, D., AND CIMIANO, P. Linking lexical resources and ontologies on the semantic web with Lemon. *The Semantic Web: Research and Applications* (2011), 245–259.

[87] MCKEOWN, K. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*.

Cambridge University Press, Cambridge, England, 1985.

[88] MELLISH, C. Using semantic web technology to support NLG case study: OWL finds RAGS. In *Proceedings of the 6th International Natural Language Generation Conference* (Stroudsburg, PA, USA, 2010), INLG '10, Association for Computational Linguistics, pp. 85–93.

[89] MELLISH, C., AND PAN, J. Finding Subsumers for Natural Language Presentation. In *Proceedings of the Internatioal Workshop on Description Logics* (Windermere, England, 2006).

[90] MELLISH, C., AND PAN, J. Natural language directed inference from ontologies. *Artificial Intelligence 172*, 10 (June 2008), 1285–1315.

[91] MELLISH, C., SCOTT, D., CAHILL, L., EVANS, R., PAIVA, D., AND REAPE, M. A Reference Architecture for Natural Language Generation Systems. *Natural Language Engineering 12*, 1 (2006), 1–34.

[92] MELLISH, C., AND SUN, X. Natural language directed inference in the presentation of ontologies. In *Proceedings of the 10th European Workshop on Natural Language Generation* (2005).

[93] MELLISH, C., AND SUN, X. The Semantic Web as a Linguistic Resource: Opportunities for Natural Language Generation. *Knowledge-Based Systems 19*, 5 (Sept. 2006), 298–303.

[94] MIEZITIS, M. Generating lexical options by matching in a knowledge base. csri-217. Tech. rep., Dept. of Computer Science, University of Toronto, Toronto, 1988.

[95] MONTIEL PONSODA, E., AGUADO DECEA, G., A., G. P., AND PETERS, W. Enriching ontologies with multilingual information. *Natural Language Engineering 17*, 3 (2011).

[96] MOORE, J., AND PARIS, C. Planning Text for Advisory Dialogues: Capturing Intentional and Rhetorical Information. *Computational Linguistics 19*, 4 (1993), 651–694.

[97] MURRAY, G., CARENINI, G., AND NG, R. Generating and validating abstracts of meeting conversations: A user study. In *Proceedings of the 6th International Natural Language Generation Conference* (Stroudsburg, PA, USA, 2010), INLG '10, Association for Computational Linguistics, pp. 105–113.

[98] NIRENBURG, S., AND NIRENBURG, I. A Framework for Lexical Selection in Natural Language Generation. In *Proceedings of COLING-88* (Budapest, 1988).

[99] NOGIER, J.-F., AND ZOCK, M. Lexical Choice as Pattern Matching. In *Current Directions in Conceptual Structures Research*, T. Nagle, Nagle, and Eklund, Eds. Ellis Horwood, 1992.

[100] O'DONNELL, M., MELLISH, C., OBERLANDER, J., AND KNOTT, A. ILEX: An architecture for a dynamic hypertext generation system. *Natural Language Engineering 7* (2001), 225–250.

[101] PHILLIPS, J. Generation of text from logic formulae. *Machine Translation 8*, 4 (1993), 209–235.

[102] POLGUÈRE, A. Pour un modèle stratifié de la lexicalisation en génération de texte. *t.a.l 39*, 2 (1998), 57–76.

[103] POWER, R. Towards a generation-based semantic web authoring tool. In *Proceedings of the 12th European Workshop on Natural Language Generation* (Stroudsburg, PA, USA, 2009), ENLG '09, Association for Computational Linguistics, pp. 9–15.

[104] POWER, R., SCOTT, D., AND EVANS, R. What You See Is What You Meant: Direct Knowledge Editing with Natural Language Feedback. In *Proceedings of the 13th Biennial European Conference on Artificial Intelligence (ECAI 98)* (Brighton, UK, 1998).

[105] POWER, R., AND THIRD, A. Expressing OWL axioms by English sentences: Dubious in theory, feasible in practice. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters* (Stroudsburg, PA, USA, 2010), COLING '10, Association for Computational Linguistics, pp. 1006–1013.

[106] REITER, E. *Generating Appropriate Natural Language Object Descriptions*. PhD thesis, Aiken Computation Lab, Harvard University, 1990.

[107] REITER, E. Has a Consensus NL Generation Architecture Appeared, and Is It Psycholinguistically Plausible? In *Proceedings of the Seventh International Workshop on Natural Language Generation (INLGW)* (Kennebunkport, Maine, USA, 1994), pp. 163–170.

[108] REITER, E., AND DALE, R. *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge, 2000.

[109] RÖSNER, D., AND STEDE, M. Customizing RST for the Automatic Production of Technical Manuals. In *Aspects of Automated Natural Language Generation*, R. Dale, E. Hovy, D. Rösner, and O. Stock, Eds. Springer Verlag, Berlin, 1992.

[110] SCHÜTTE, N. Generating natural language descriptions of ontology concepts. In *Proceedings of the 12th European Workshop on Natural Language Generation* (2009), pp. 106–109.

[111] SCHWITTER, R., AND TILBROOK, M. Controlled natural language meets the semantic web. In *Proceedings of the Australasian Language Technology Workshop* (2004), vol. 2004.

[112] SMART, P. Controlled natural languages and the semantic web. Technical report, University of Southampton, July 2008.

[113] SMEDT, K. D., HORACEK, H., AND ZOCK, M. Architectures for Natural Language Generation: Problems and Perspectives. In *In Trends in Natural Language Generation: An Artificial Intelligence Perspective* (1996), Springer-Verlag, pp. 17–46.

[114] STEDE, M. *Lexical Semantics and Knowledge Representation in Multilingual Text Generation*. Kluwer Academic Publishers, Dordrecht, 1999.

[115] STEVENS, R., MALONE, J., WILLIAMS, S., POWER, R., AND THIRD, A. Automating generation of textual class definitions from OWL to English. *Journal of Biomedical Semantics 2*, 2:S5 (2011).

[116] SUN, X., AND MELLISH, C. Domain independent sentence generation from RDF representations for the Semantic Web. In *Combined Workshop on Language-Enabled Educational Technology and Development and Evaluation of Robust Spoken Dialogue Systems, European Conference on AI, Riva del Garda, Italy* (2006).

[117] SUN, X., AND MELLISH, C. An experiment on "free generation" from single RDF triples. In *Proceedings of the 11th European Workshop on Natural Language Generation* (2007), Association for Computational Linguistics, pp. 105–108.

[118] WANG, C., XIONG, M., ZHOU, Q., AND YU, Y. PANTO: A Portable Natural Language Interface to Ontologies. In *The Semantic Web: Research and Applications*, E. Franconi, M. Kiefer, and W. May, Eds., vol. 4519 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2007, pp. 473–487.

[119] WANNER, L. Lexical Choice in Text Generation and Machine Translation. *Machine Translation 11*, 1–3 (1996).

[120] WANNER, L. *Exploring Lexical Resources for Text Generation in a Systemic Functional Language Model*. PhD thesis, Saarland University, Saarbrücken, 1997.

[121] WEAL, M., ALANI, H., KIM, S., LEWIS, P., MILLARD, D., SINCLAIR, P., DE ROURE, D., AND SHADBOLT, N. Ontologies as facilitators for repurposing web documents. *International Journal of Human-Computer Studies 65*, 6 (June 2007), 537–562.

[122] WHITE, M. Statistical Natural Language Generation, 2010.

[123] WILCOCK, G. Pipelines, Templates and Transformations: XML for Natural Language Generation. In *Proceedings of the First NLP and XML Workshop* (Tokyo, 2001), pp. 1–8.

[124] WILCOCK, G., AND JOKINEN, K. Generating responses and explanations from rdf/xml and daml+oil. *Knowledge and Reasoning in Practical Dialogue Systems, IJCAI 2003* (2003), 58–63.

[125] WILKS, Y., AND BREWSTER, C. Natural Language Processing as a Foundation of the Semantic Web. *Foundations and Trends in Web Science 1*, 3–4 (2009), 199–327.

[126] WILLIAMS, S., AND POWER, R. Grouping axioms for more coherent ontology descriptions. In *Proceedings of the 6th International Natural Language Generation Conference* (Stroudsburg, PA, USA, 2010), INLG '10, Association for Computational Linguistics, pp. 197–201.

[127] XU, F., ADOLPHS, P., USZKOREIT, H., CHENG, X., AND LI, H. Gossip Galore: A Conversational Web Agent for Collecting and Sharing Pop Trivia. In *Proceedings of the International Conference on Agents and Artificial Intelligence (ICAART)* (2009), pp. 115–122.

[128] ZHANG, X., CHENG, G., AND QU, Y. Ontology summarization based on RDF sentence graph. In *Proceedings of the 16th international conference on World Wide Web* (New York, NY, USA, 2007), WWW '07, ACM, pp. 707–716.