Making Sense of schema.org with WordNet

Csaba Veres
Department of Information Science and Media Studies
The University of Bergen, Norway
csaba.veres@uib.no

Abstract

The *schema.org* initiative was designed to introduce machine readable metadata into the World Wide Web. This paper investigates conceptual biases in the *schema* through a mapping exercise between *schema.org* types and WordNet synsets. We create a mapping ontology which establishes the relationship between *schema* metadata types and the corresponding everyday concepts. This in turn can be used to enhance metadata annotation to include a more complete description of knowledge on the Web of data.

1 Introduction

Schema.org is an initiative to introduce machine readable metadata into HTML Web pages. It was launched on June 2, 2011, under the auspices of a consortium consisting of Google, Bing, and Yahoo!. The *schema.org* web site initially described the project as one that "provides a collection of schemas, i.e., html tags, that webmasters can use to markup their pages in ways recognized by major search providers .... making it easier for people to find the right web pages." (schema.org web site, 2011). The incentive for using the schema was that web sites that contained markup would appear with informative details in search results which in turn enables people to judge the relevance of the site more accurately. This could lead to higher user engagement and higher search ranking, which is the ultimate incentive for web masters.

The initial release contained 297 classes and 187 relations, but by 2016 had grown to 638 classes and 965 relations (Guha et al., 2016). It is important to note, however, that the expansion of the *schema* consists entirely in adding subclasses and properties to the core classes through the allowed extension mechanism[1]. From the outset the immediate sub classes of *Thing* were stiulated as *Action, CreativeWork, Event, Intangible, Organization, Person, Place* and *Product*. These high level conceptual divisions with their implicit ontological commitments are not, and never were open to discussion.

(Guha et al., 2016) explain that the primary driving force behind the design of the *schema*, and ultimately the reason for its success, was its simplicity. Previous efforts to introduce large scale metadata failed, in part because each standard was too narrow in terms of domain coverage. The result was too many standards for too few applications. On the other hand the *schema* offered a single, unified and broad vocabulary that could be used across several verticals and promised a benefit for perhaps the most important driving force, search rankings. As a part of this simplicity, the *schema* taxonomy and classes were intended more as an "organisational tool to help browse the vocabulary" than a definitive ontology of world (Guha et al., 2016). In other words, the *schema* was designed as an intuitive set of metadata classes that could be used to describe the majority of items people would search for on the Web.

Together these factors ensured that the *schema* has enjoyed a significant amount of success. (Guha et al., 2016) report that in a sample of 10 billion web pages, 31.3% of the pages had *schema.org* markup, a growth of 22% from a year earlier. The markup is used by many different data consumers for various tasks involving enhanced search results (rich snippets), populating the Google Knowledge Graph, exchange of transaction details in email, support for automatic formatting of recipes, reviews, etc., and advanced search features in Apple's Siri. The fifteen most popular implemented classes were *WebSite, SearchAc-*

[1] https://is.gd/HdnHkp

tion, WebPage, Product,ImageObject, Person, Offer, BlogPosting, Organization, Article, PostalAddress, Blog, LocalBusiness, AggregateRating, WPFooter. Many of these refer to elements of the web page itself rather than the content. The top fifteen content bearing classes were *Product, ImageObject, Person, Offer, Organization, PostalAddress, LocalBusiness, AggregateRating, CreativeWork, Review, Place, Rating, Event, GeoCoordinates*, and *Thing*. These are sun types of *Product, CreativeWork, Person, Intangible, Organization, Place,* and *Event*. Although the coverage was intended to be broad, it is clear that the use of the *schema* covers its range of types well, but that the types favour a particular view of web content, in the interests of the search providers.

The motivation for this paper was to try and characterize the conceptual biases of the *schema* top level categories, by mapping the types to their corresponding meanings in WordNet. To the extent that we believe WordNet captures the ontological commitments inherent in human language, it should provide insights about where the two conceptualisations diverge. The further aim, however, is to use the mappings to enrich the valuable human provided metadata towards the aim of providing general but rich meaning annotations to a large portion of Web content.

It is important to note that we are not advocating WordNet as a *gold standard* for ontologies and knowledge representation. On the contrary, we agree with (Hirst, 2004) who argues that WordNet contains modeling decisions which differentiate it from formal ontologies. As an example, there are cases where synsets have overlapping hyponyms whereas ontologies have disjoint subclasses. Consider the first noun sense of *mistake: {mistake, error, fault}* which includes the following hyponyms (among others): *{slip, slip-up, miscue, parapraxis}, {oversight, lapse}, {faux pas, gaffe, solecism, slip, gaucherie}, and {failure}*. A single act can be both a *slip* and a *faux pas*. The first implies the act was inadvertent, and the second that it possibly had a social component such as a mistake in etiquette. A *lapse* is also a *slip*, but it involves some sort of forgetfulness or inattention on top of the mere *slip*. A lapse can also be a *faux pas*, of course. If the *faux pas* is sufficiently severe, it can become a complete *failure*. These hyponyms contain more information that that they are a *kind-of mistake*, they also contain information about likely causes and implications, and these can be overlapping. Nevertheless, our interest is that people **do** consider these as kinds of mistake in everyday discourse. For the same reason we think it is beside the point to try and restrucutre WordNet by some formal methodology such as DOLCE (Gangemi et al., 2003a). We are interested here in intuitive relations, not formal ones.

2 The WordNet Mappings

The mapping involved two stages. First the *schema.org* types were aligned with WordNet synsets, while retaining the structure of the *schema*. This stage can be seen as adding information to the *schema*, namely, the corresponding WordNet synsets. Then, a new hierarchy of concepts was constructed from the synsets involved in the mapping. That is, by promoting the mapped synsets to be the central classes, we could get a better idea what sorts of concepts are in the *schema*, in relation to the WordNet taxonomy.

In order to distinguish between the concepts in the two taxonomies, WordNet names will be prefixed with *wn:* and the *schema* with the prefix *schema:*. In addition when necessary the WordNet name will be qualified with part of speech and sense tag, as in *wn:dog#n#1*.

To summarize, we constructed two artefacts at the end of the process:

- The WordNet to schema.org mapping ontology. This retains the *schema* class structure. The mappings were manually constructed and available on GitHub[2].

- The WordNet taxonomy for the synsets that have been mapped to the schema. This shows an alternative taxonomy of the words in the *schema*.

2.1 The Mapping Ontology

In this ontology the original *schema.org* taxonomy was retained, and the WordNet synsets were simply inserted into this taxonomy. In fig. 1 we see some example mappings, showing schema:Beach mapped to wn:beach. Since schema:Beach is a subtype of schema:CivicStructure, by implication so too is wn:beach. Similarly, the other WordNet synsets in the example become subclasses of schema:CivicStructure through their respective

[2] `https://is.gd/XF0bJe`

alignments. The mapping provides the immediate benefit that web sites which contained any of the WordNet synsets in the alignment, could automatically be connected to their corresponding *schema* types. This suggests a method for automatic metadata creation, which will be dicussed subsequently.

Notice that the mapping is not straightforward and in this example synsets of quite distinct types are grouped under the one *schema* type. For example wn:bus_terminal <is-a> wn:facility, wn:cinema <is-a> wn:theater <is-a> wn:building, and a wn:parking_lot <is-a> wn:tract,piece_of_land. Yet they all map to subclasses of schema:CivicStructure.

The second taxonomy was created precisely to reveal the *schema* conceptualisation in terms of the WordNet hierarchy. In other words, *"what IS a schema:CivicStucture in everyday language?"*

2.2 The WordNet Ontology

The full WordNet hypernym tree is quite deep, and quickly leads to a very complex taxonomy. For this reason we made use of a simple tool which uses an algorithm to eliminate low information nodes from a taxonomy (Veres et al., 2013). The algorithm prunes the tree by counting the number of outward links at each node, and eliminating any node that has fewer than a certain number of (user specified) hyponyms. When this is performed on every node in the graph, what remains is a number of intermediate synsets which are the maximally informative hypernyms of any leaf node. In the graphs reported here, the lower threshold was set at 3. The tool essentially implements the algorithm used by (Stoica and Hearst, 2004), but our interface has the advantage that the parameters can be dynamically adjusted and visually inspected to give the most intuitively pleasing result. A similar procedure was followed in (Izquierdo et al., 2006) to identify basic level concepts. Our work differs in that we do not distinguish between nodes above the basic threshold.

A part of the inferred hierarchy involving wn:beach is shown in figure 2. Note that wn:beach is a sibling of wn:mountain, whereas the *schema* choice to model the *civic structure* aspect of *beach* puts them in different subclasses; schema:Beach is a schema:CivicStructure while schema:Mountain is a schema:Landform. However, since wn:beach is a hyponym of wn:geological_formation, which in turn is an equivalent class of schema:LandForm, it could be inferred that schema:Beach could also be a schema:LandForm. The benefit of the alignment is that a new and sensible *schema* type could be added to any markup involving *beach*. Figure 3. shows how the WordNet hierarchy connects wn:beach to schema:Landform and potentially other subclasses. A web site about a geographical area with mountains and beaches could then be appropriately annotated.

Looking at the taxonomy itself, we can see what kind of WordNet synsets appear in the *schema*. The major division in fig. 4 is between wn:physical_entity and wn:abstraction, which is an ontological distinction that is typically considered fundamental (e.g. (Niles and Pease, 2001), (Gangemi et al., 2003b)). On this view the *schema* describes the world as populated by *physical entities* and *abstractions*, where the *physical entities* are predominantly *objects*, and *abstractions* are diverse sorts of *events* or *roles* which the entities engage in. For example wn:measure is *how much there is of something you can quantify*, and wn:state is *the way something is with respect to its attributes*. Other sub types of wn:abstraction, like wn:organization and wn:tourist_attraction apply to concepts that are typically human centered, functional collections of objects (Wierzbicka, 1984). Wierzbicka argues that putatively taxonomic concept hierarchies are in fact the majority of the time made up of a mixture of supercategory types, with the most prominent two being taxonomic and functional. (Pustejovsky, 1991) draws a similar distinction with the mechanism of *formal* and *telic* roles in his lexical structures.

The ontological commitment adopted by *schema.org* becomes clear if we compare the two taxonomies. The *schema* divides schema:Thing into: schema:CreativeWork, schema:Event, schema:Intangible, schema:Organization, schema:Person, schema:Place, and schema:Product. The focus is immediately on the functional categories: telic roles dominate the top level categories of the *schema*, and physical entities are sub types of these abstractions.

The most obvious example of a top-level purely functional type is schema:Product. Almost anything can be a *product*, and there is no property which *products* have in common except the telic

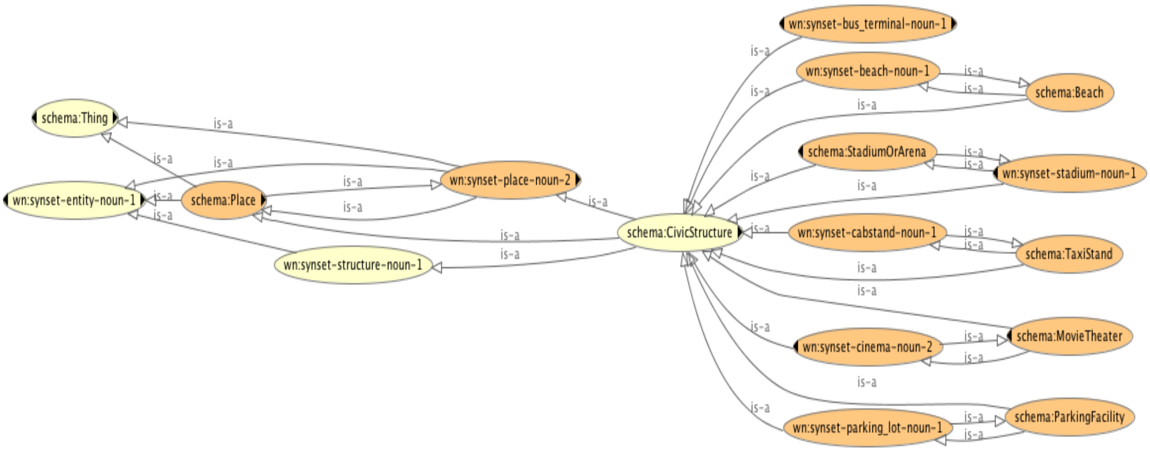

Figure 1: Example mappings between WordNet and schema.org, for the corresponsing concepts *beach*. The ovals in darker shading represent concepts which have equivalent classes in the two namespaces.

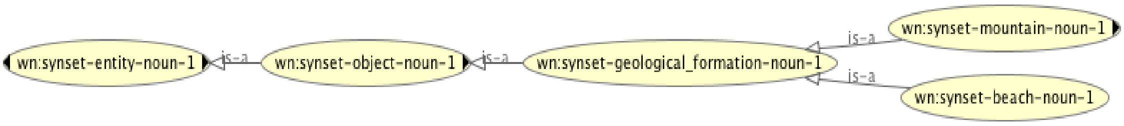

Figure 2: Part of the WordNet taxonomy

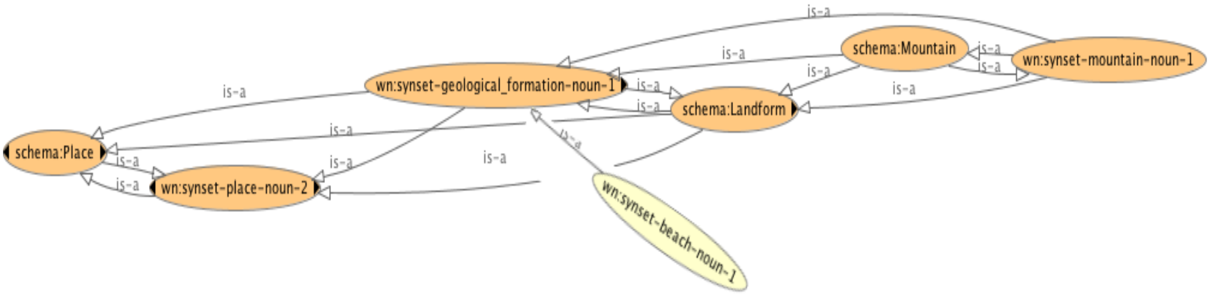

Figure 3: wn:beach inherits schema:Landform

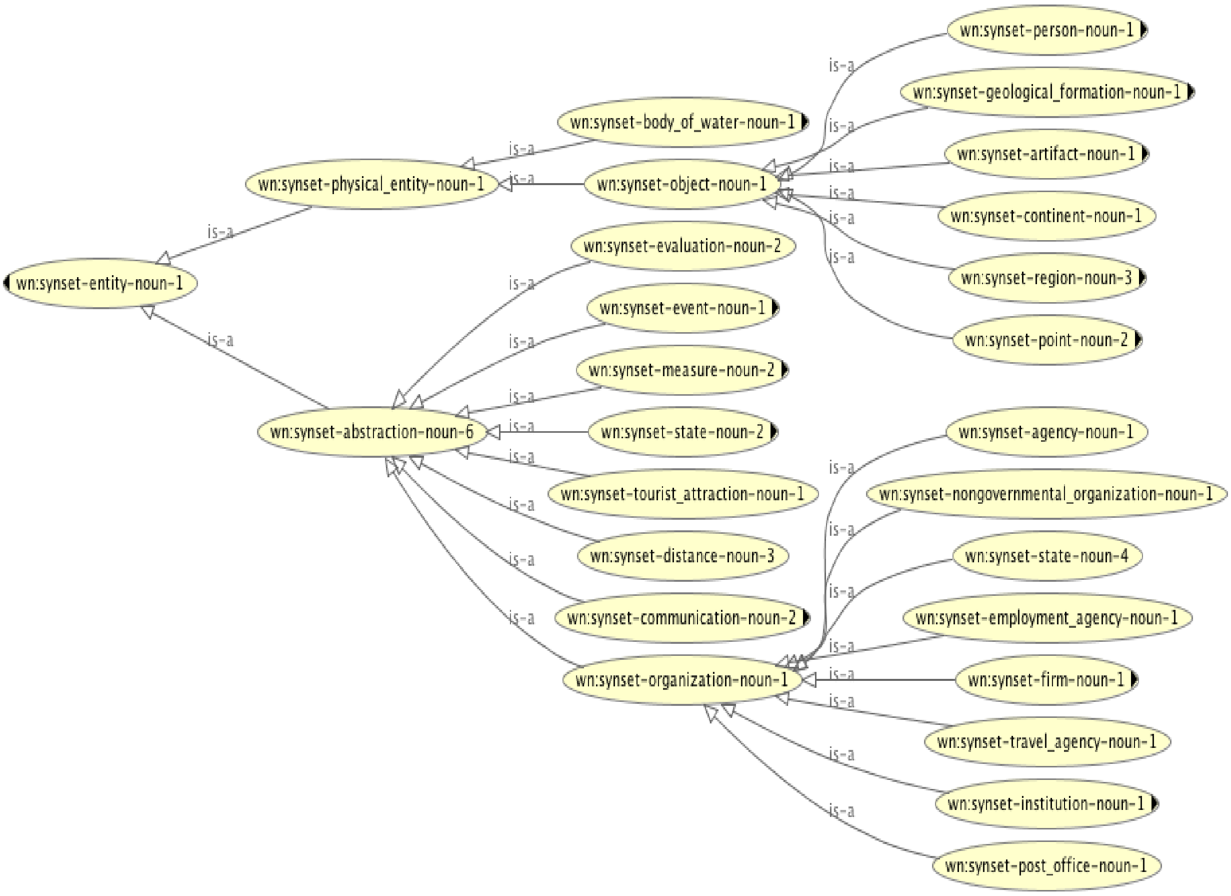
wn:synset-entity-noun-1
wn:synset-physical_entity-noun-1
wn:synset-abstraction-noun-6
wn:synset-body_of_water-noun-1
wn:synset-object-noun-1
wn:synset-person-noun-1
wn:synset-geological_formation-noun-1
wn:synset-artifact-noun-1
wn:synset-continent-noun-1
wn:synset-region-noun-3
wn:synset-point-noun-2
wn:synset-evaluation-noun-2
wn:synset-event-noun-1
wn:synset-measure-noun-2
wn:synset-state-noun-2
wn:synset-tourist_attraction-noun-1
wn:synset-distance-noun-3
wn:synset-communication-noun-2
wn:synset-organization-noun-1
wn:synset-agency-noun-1
wn:synset-nongovernmental_organization-noun-1
wn:synset-state-noun-4
wn:synset-employment_agency-noun-1
wn:synset-firm-noun-1
wn:synset-travel_agency-noun-1
wn:synset-institution-noun-1
wn:synset-post_office-noun-1
is-a

Figure 4: Part of the WordNet taxonomy from SynsetTagger

role that they are "made available for sale". One can sell a sewing needle or a Saturn V rocket. Actually the situation is even more complicated because Products don't even have to be individuated "things". The documentation of schema:Product reads: "a pair of shoes; a concert ticket; the rental of a car; a haircut; or an episode of a TV show streamed online".

The fact that there are in fact a number of functional categories at the highest level helps explain the strange tangle of types at the lower levels of the hierarchy, where many different kinds of things (in the formal, taxonomic sense) can appear if they serve particular functions. To see how this becomes problematical, consider the common functional category *weapon* which can include items such as *crossbow, flamethrower, gun, knife, poison gas, anthrax bacillus, novichok, boomerang, and hydrogen bomb*. Clearly as individual objects these would have quite different sets of properties. The problem for the *schema* is that different *formal* objects are forced to coexist as siblings in a taxonomy dominated by *telic* roles. This results in examples such as schema:Beach having opening hours, schema:Continent with a telephone number and review, and other *strange and wonderful things*. One is forced to assume that schema:Beach was designated as a schema:CivicStructure, for example, because the emphasis is on the facilities available at the beach, not the beach itself.

The inclusion of telic roles such as schema:Product at such a high level of generality has the additional consequence that the *schema* does not contain a type which corresponds to the simple notion of a *physical object*. There is no option in *schema.org* for the structured markup of cars, boats, computer chips, barbells, antiques, or any of the other hundred million human artefacts ancient and modern, except as a "Product", because the schema lumps these into the class of "sellable things". Neither does there seem to be any proper place for natural objects like *cats* or *dogs*[3] or *tree* amd*forest*, which simply have no place.

Finally it should be noted that the hierarchy in WordNet does also include purely functional types among its hypernyms. For example in the *weapon* example above we see that wn:gun is-a wn:weapon is-a wn:object. George Miller (in (Fellbaum, 1998)) explains that this was perhaps an unfortunate problem that might have been avoided had the importance of Wierzbicka's work been realized earlier. However, the structure of WordNet ensures that, whenever such a confusion exists, the formal properties of the word are still recorded. One mechanism is that words can appear in more than one hierarchy. For example *anthrax bacillus* is both a wn:microorganism, and a wn:weapon. Another possibility is that words with both roles are listed twice. For example wn:chicken#n#1 <is-a> wn:meat#n#1, and wn:chicken#n#2 <is-a> wn:bird#n#1. The *schema* only offers one choice for the poor chicken, schema:MenuSection.

3 Finding correct mappings

There are a number of potential pitfalls in defining appropriate mappings between the two taxonomies. One of the most important is to avoid introducing unwanted inferences from the semantics of the mapping axioms. A prevalent example of this is the use of owl:sameAs to represent equivalence between individuals, or classes in OWL-Full. owl:sameAs asserts full equivalence between the individuals such that all of their properties are automatically shared, even though most commonly this is not the desired consequence (Halpin et al., 2010). To avoid this problem we used the weaker owl:equivalentClass axiom, which does not imply complete equality. What is required instead is the weaker condition that every instance of one class must also be an instance of the other.

Even with a weaker semantics we found that equivalent classes could not always be found. One reason is that schema.org includes concepts which involve various sorts of compounding of simple concepts, and WordNet contains only common, lexicalized compounds. For example LandmarksOrHistoricalBuildings is a compound concept that includes any kind of general *landmark* as well as the specific concept of *buildings with historical significance*. There is no such lexical entry in English. Most likely there is no such compound in any language, because the concept is un-natural, mixing different levels of generalization. It is analogous to a concept for *toys or teddy bears*.

There are also more acceptable compounds like schema:CivicStructure which is "a public structure such as a town hall or concert hall". This is of course a perfectly acceptable compound, which

[3] the search facility suggests schema:AnimalShelter

happens not to be in WordNet. In every case that an acceptable WordNet compound could not be found, we decided to make the *schema.org* concept a subclass of one or more WordNet synsets that captured part of the compound. For the above example of schema:CivicStructure, the obvious superclass is wn:structure#n#1.

Sometimes the compound nature of the *schema* terms is hidden. For example the terms that are subclasses of schema:LocalBusiness are a mixed group of explicit compounds (e.g., schema:MovingCompany, schema:IceCreamShop) and implicit compounds (e.g., schema:Electrician, schema:Locksmith, schema:HousePainter). That is, schema:Electrician is really meant to be something like "ElectricianBusiness" and not just "Electrician". The compound schema:HousePainter is even more complicated because it has an exact match in wn:house_painter#n#1, but in fact schema:HousePainter is really meant to be a HousePainterBusiness, so the exact match is illusory. The important modelling decision is whether or not to reintroduce the hidden compound in mapping to WordNet. That is, should schema:Electrician be regarded in its ordinary word sense as "a person who is an electrician", or should it be modelled as an "electrician business"? In other words, these concepts could simply be declared as subclasses of wn:place_of_business to maintain the intended interpretation in the *schema*. The most flexible solution was to declare an equivalent class relation between schema:Electrician and the person interpretation in WordNet, wn:electrician#n#1. This choice captures the notion that electricians are people. However it is also possible to infer that wn:electrician is a wn:place_of_business, as shown in Figure 5.

There is a small set of *schema.org* types for which we did not establish mappings. One group involved technical compounds describing the structure of web pages with terms like schema:AboutPage and schema:CheckoutPage. These are all subtypes of schema:WebPage, for which we did define a mapping. The second group was the primitive data types, schema:DataType which are not part of the main taxonomy subsumed by schema:Thing.

4 Using the WordNet Mappings

The practical motivation for mapping the *schema* to WordNet was to enrich the metadata that can be assigned to concepts in a web page. We have already seen this in examples such as *beach*. A secondary motivation was to make it easier for web masters to find the *schema* types without knowing anything about its structure. We have already developed a prototype of a tool in which the user can highlight any word in text, nominate its corresponding synset, and the application will attempt to guess the correct *schema* type. Consider the following example scenario.

There is a geological landmark called the Jenolan Caves in the Blue Mountains, Australia. Suppose a web master wanted to mark up the web site for Jenolan Caves. A quick search will reveal that there is no matching type in the *schema* for caves. Using the WordNet mappings it is possible for the designer to find the most appropriate types, without any knowledge of the *schema*. The synset wn:cave is a wn:geological_formation, which in turn maps to schema:Landform. However, the mapping ontology can also suggest additional useful classifications. The coordinate terms of wn:cave contain some terms which **are** defined in the *schema*, including our old friend *beach*. Recall that wn:beach is mapped to schema:CivicStructure through schema:Beach (see Figure 6). Thus Jenolan Caves could be marked with both *schema* types, and the properties of the facilities at the premises could be specified. Of course the annotation effort does not have to stop there. Since the WordNet synset is available, it can also be included in the markup, which in turn enables the markup to be used with a huge number of mappings to other resources[4].

While this process is currently being performed through our prototype tool where users specify the disambiguated sense (Veres and Elseth, 2013), this does not necessarily have to be performed manually. With sufficiently accurate disambiguation methods, any web page could be automatically annotated with *schema* and WordNet metadata. This would be useful for any downstream task including the construction of knowledge graphs, as previously mentioned.

The Jenolan Caves example requires the ability to declare multiple types. The original syntax for

[4] `https://wordnet.princeton.edu/related-projects`

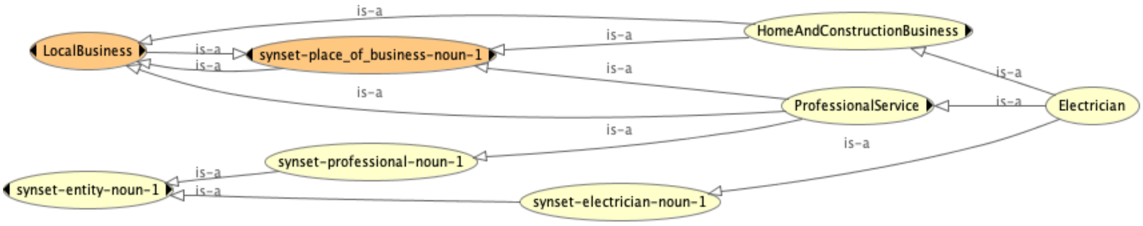
is-a
HomeAndConstructionBusiness
LocalBusiness
synset-place_of_business-noun-1
ProfessionalService
Electrician
synset-professional-noun-1
synset-entity-noun-1
synset-electrician-noun-1

Figure 5: Electrician as both a person (wn:electrician#n#1) and place of business (wn:place_of_business#n#1).

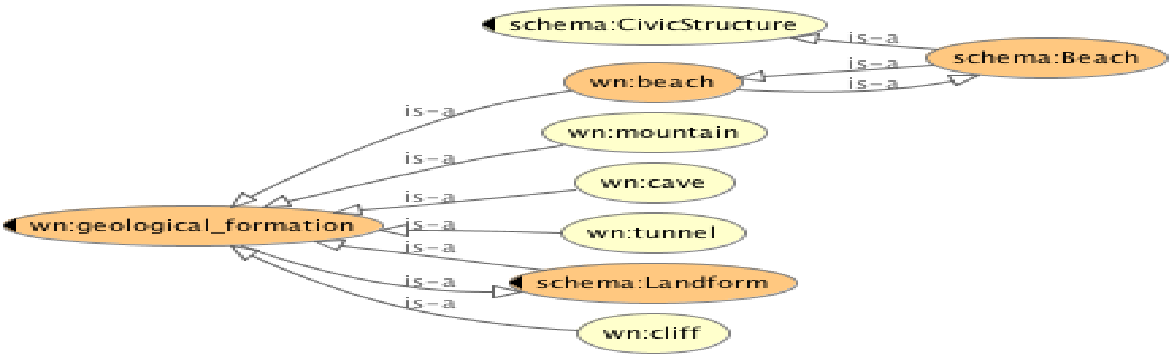

Figure 6: Mapping "cave" to schema.org types

the *schema*, *microdata* is not able to express multiple types. The recommendation therefore is to use *rdf-a*[5] or *json-ld*[6] which are inherently built to express multiple types from any vocabulary.

5 Conclusion

We proposed a method for evaluating the conceptual bias of *schema.org* by comparing the type terms against their usage in everyday language as stipulated in WordNet. The observation is that *schema.org* favours the markup of web sites promoting goods, services, and locations fulfilling some human centred need. This then results in the observed data that the majority of web sites which contain *schema.org*, are about products and goods and services. If search rankings favour sites with markup, and if most markup is about goods and services, then search results will come to favour goods and services. Anecdotally, this could be one factor for why it is sometimes easier to find where to buy something rather than information about the thing itself. The bias diminishes the potential for providing a rich source of general semantic metadata on the web, for use in diverse use cases.

We argued that the *schema* needs types that describe a more neutral view of the world, for example artefacts, to describe *things* independently of the *roles* they can play. A metadata specification should be able to annotate a *chicken* as a kind of *bird* as well as a kind of *food*.

Our suggestion to include WordNet mappings into the markup effort is one way to sneak more general markup into the annotation process. The requirement is that multiple types must be a standard feature of the annotation, with different types describing different aspects of the item. A *car* is an artefact designed for locomotion, but can also acquire its role as a *product* if it is put up for sale. This addition would not compromise people who want to advertise their products. In fact, it would give them more freedom to express physical properties of their products like size, construction material, origin, and so on.

In summary, we used WordNet as a standard representation of everyday word use, to provide clarity to the types proposed in *schema.org*. We proposed a method to help people mark up Web sites that do not fit neatly into the service oriented world view, by enabling them to annotate their contribution to world knowledge as broadly as possible. This is clearly of benefit to all users who see the web as a vehicle for disseminating informative structured data as freely as possible.

[5] `https://rdfa.info/`

[6] `https://json-ld.org/`

References

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Aldo Gangemi, Nicola Guarino, Claudio Masolo, and Alessandro Oltramari. 2003a. Sweetening wordnet with dolce. *AI Magazine*, 24:13–24.

Aldo Gangemi, Nicola Guarino, Claudio Masolo, and Alessandro Oltramari. 2003b. Sweetening wordnet with dolce. *AI Mag.*, 24(3):13–24, September.

RV Guha, Dan Brickley, and Steve Macbeth. 2016. Schema.org: evolution of structured data on the web. *Communications of the ACM*, 59:44–51.

Harry Halpin, Patrick J. Hayes, James P. McCusker, Deborah L. McGuinness, and Henry S. Thompson. 2010. When owl:sameas isn't the same: An analysis of identity in linked data. In Peter F. Patel-Schneider, Yue Pan, Pascal Hitzler, Peter Mika, Lei Zhang, Jeff Z. Pan, Ian Horrocks, and Birte Glimm, editors, *The Semantic Web – ISWC 2010*, pages 305–320, Berlin, Heidelberg. Springer Berlin Heidelberg.

Graeme Hirst, 2004. *Ontology and the Lexicon*, pages 209–229. Springer Berlin Heidelberg, Berlin, Heidelberg.

Rubén Izquierdo, Armando Suárez, German Rigau, and Ixa Nlp. 2006. Exploring the automatic selection of basic level concepts. In *International Conference Recent advance in Natural Language Processing*.

Ian Niles and Adam Pease. 2001. Towards a standard upper ontology. In *Proceedings of the International Conference on Formal Ontology in Information Systems - Volume 2001*, FOIS '01, pages 2–9, New York, NY, USA. ACM.

James Pustejovsky. 1991. The generative lexicon. *Comput. Linguist.*, 17(4):409–441, December.

Emilia Stoica and Marti A. Hearst. 2004. Nearly-automated metadata hierarchy creation. In *Proceedings of HLT-NAACL 2004: Short Papers*, HLT-NAACL-Short '04, pages 117–120, Stroudsburg, PA, USA. Association for Computational Linguistics.

Csaba Veres and Eivind Elseth. 2013. Schema.org for the semantic web with madame. In *Proceedings of the I-SEMANTICS 2013 Posters & Demonstrations Track, Graz, Austria, September 4-6, 2013*, pages 11–15.

Csaba Veres, Kristian Johansen, and Andreas Opdahl. 2013. Synsettagger: A tool for generating ontologies from semantic tags. In *Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics*, WIMS '13, pages 16:1–16:10, New York, NY, USA. ACM.

Anna Wierzbicka. 1984. Apples are not a "kind of fruit": the semantics of human categorization. *American Ethnologist*, 11(2):313–328.