Extracting Semantics of Places from User Generated Content

Heidelinde Hobel
Technische Universität Wien
Department for Geodesy and Geoinformation
Gusshausstrasse 27-29
1040 Vienna, Austria
hobel@geoinfo.tuwien.ac.at

Paolo Fogliaroni
Technische Universität Wien
Department for Geodesy and Geoinformation
Gusshausstrasse 27-29
1040 Vienna, Austria
paolo@geoinfo.tuwien.ac.at

Abstract

Next generation Geographic Information Systems should support place-based searches. The notion of place is a vague concept that strictly relates to human conceptualization of space. We regard places as cognitive regions affording activity opportunities and present a computational workflow to populate the model with information from User Generated Content available on the Web. An algorithmic realization is provided that relies on the Resource Description Framework along with a real example derived by an implementation of the workflow that relies on OpenStreetMap and TripAdvisor data.

Keywords: Spatial Search, Natural Language Processing, Model of Place, Semantic Web.

1 Introduction

Modern Geographic Information services (e.g., Google Maps) mostly support searches based on name and category matching as well as on spatial relations among geographic features (e.g., the Opera House in Vienna, restaurants nearby Vienna). An important category of search that is not yet supported concerns the retrieval of places. One major reason hindering such type of search is the lack of a cognitively plausible model for places that is capable of capturing human understanding of such a fuzzy term.

Drawing upon the taxonomy for geographic regions proposed in [19] and on affordance theory [8, 9], we regard a place as a *cognitive region*. More specifically, as a region of space conceptualized as a whole by people because of the activities it affords.

As a source for activity information we suggest to exploit User Generated Content (UGC) like geo-logs, travel social media (e.g., TripAdvisor[1]) and place review forums. Arguably, these are suitable sources to extract the coveted information, as they convey human conceptualization of places in the form of unstructured textual representations of cognitive regions.

In the spirit of the Semantic Web [4], this paper presents on-going work aimed at building a vocabulary to model places in terms of the activities they afford. A preliminary, simplified vocabulary is presented and a computational workflow to populate the model from UGC is described. The idea of integrating activities for place reference systems is a hot research topic [21]. We focus on an algorithmic approach that exploits Natural Language Processing tools to map unstructured text onto the proposed semantic model.

The remainder of this paper is structured as follows. Section 2 reviews related work. The semantic model is introduced in Section 3. Section 4 outlines the computational workflow and describes the algorithmic approach. Section 5 concludes the work, stress the main limitations of the current approach and shortly discuss further improvements and extensions.

[1] http://www.tripadvisor.com/

2 Related Work

The semantic of place and its representation within Geographic Information Systems has recently become a prominent research area [10]. For a thorough review of the state of the art on place extraction techniques refer to [23].

To enable a cognitive view in information retrieval, common-sense knowledge bases for information retrieval tasks were proposed, which are based on relationships among spatial objects [20, 24]. Based on the notion of Naïve Geography [6], a novel framework was proposed in [7] that enables qualitative spatial relation and configuration queries as a means to provide the casual user with more natural spatial search possibilities.

Besides the spatial arrangement of objects, the question how people perceive places is tightly coupled with place affordances. The term affordances refers to Gibson's theory of visual perception and designates action potentials that are recognized by an agent in its environment [8, 9]. An object only affords an action if the agent's capabilities allow for performing such an action.

Recently, much work has also focused on modelling, publishing, and consuming spatial data within the Semantic Web [15]. A glaring example is the LinkedGeoData project [22] that provides an encoding of OpenStreetMap[2] data into the Resource Description Framework (RDF). The Linked Spatial Data trend is supported by different spatial (and temporal) extensions of basic RDF and SPARQL. Some examples are the GeoSPARQL [3] and the stSPARQL [14] vocabularies and query extensions.

Natural Language Processing (NLP) has also become a prominent research topic in Geographic Information Science. The question how to model semantics of space for the retrieval from unstructured text is addressed, for example, in [2] where an ontology is presented for the processing of language concerned with space, actions in space, and spatial relationships. Kahn et al. [13] derived spatial triplets from unstructured text while Alazzawi et al. [1] concentrated on

[2] https://www.openstreetmap.org

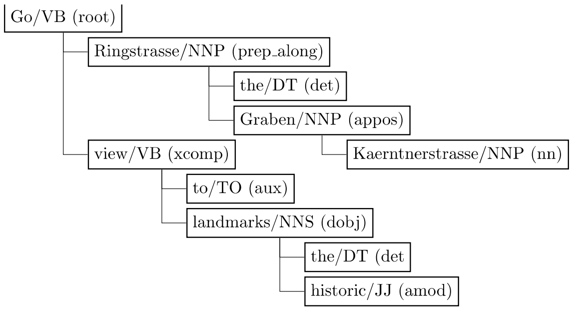
pattern mining for deriving language patterns for service identification.

Table 1: Partial list of syntactical classes and tags extracted from [18].

Tag	Definition
NN	noun, singular or mass
NNP	proper noun, singular
WP	pronoun
VB	verb
CC	coordinating conjunction
JJ	adjective
DT	determiner
WRB	adverb

Recent years witnessed an increasing proliferation of research works dealing with extraction of structured information from unstructured text. This has been largely made possible by the availability of mature NLP software like the Stanford CoreNLP toolkit [17]. This is a software suite offering tools to parse and map unstructured text onto formal structures. One of the most interesting tools in this suite is the *dependency parser* that, given a sentence, generates a so–called *dependency tree*. The nodes of a dependency tree denote the syntactical class of each word in a sentence (see Table **1** for a partial list of such syntactical classes/tags). The labeled edges represent the hierarchical structure of grammatical relations between the words. The root of a dependency tree always contains the verb of the independent clause of the sentence. The encoding of a sentence in a tree structure is done by extracting sequences of dependencies among words. Such dependencies are exactly the grammatical relations holding among different terms. If the parser is unable to narrow down the relation to a specific one, the edge is generally labeled *rel*. For example, the sentence "Go along the Ringstrasse, Kaerntnerstrasse, Graben to view the historic landmarks" is parsed onto the dependency tree shown in Figure 2, where syntactical classes and grammatical relations are both reported in the nodes.

Figure 2: Dependency tree obtained with the Stanford Dependency Parser [17] for the sentence "Go along the Ringstrasse, Kaerntnerstrasse, Graben to view the historic landmarks".

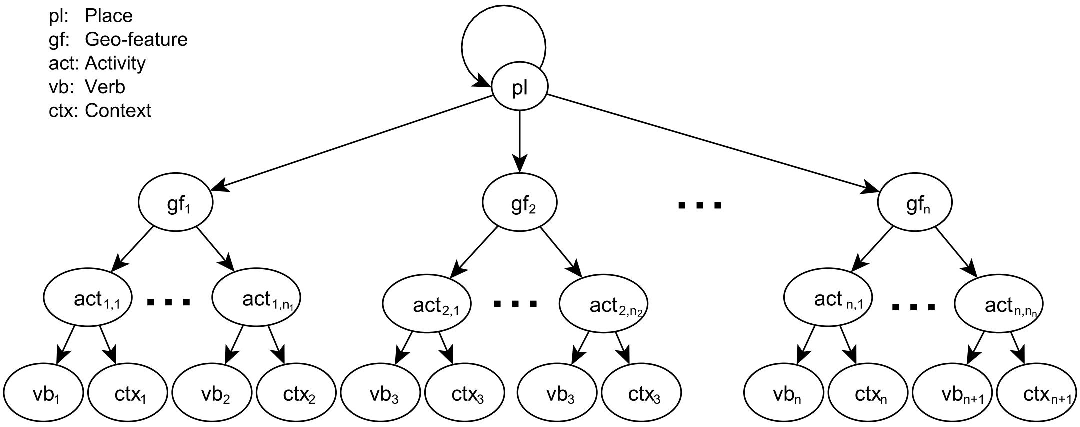


3 Mapping Semantics onto Cognitive Regions

We propose the semantic model for places (specifically, cognitive regions) depicted in
Figure 1. A place *pl* (root node in the diagram) recursively consists of other places or of an open number of geo-features that define the geographic footprint of *pl*. A geo-feature *gf* is a spatial entity (real or abstract) that possibly affords a number of different activities to be performed at or nearby the feature location. We regard an activity *act=(vb, ctx)* as a pair consisting of a verb and a context. The verb expresses the type of activity (e.g., see, eat) while the context is any piece of ancillary information that narrows down or, more generally, modifies the semantic of the activity (e.g., see historical buildings, eat an ice cream). More detailed studies on context and spatial context exist; see, for example, [11].

Note that there is no restriction on the uniqueness of verbs, contexts, and verb-context pairs. In fact, for example, an historical building (ctx_1) can be seen (vb_1) but can also be photographed (vb_2). Moreover, historical buildings can be seen and photographed at different locations.

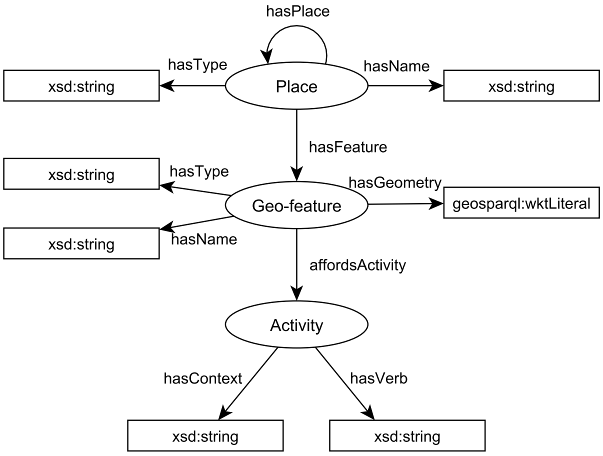
Figure 1: Abstract semantic model of a place. A place consists of one or more places or geo-features. Geo-features afford for activities. Activities consist of a verb in a given context.



According to the proposed semantic model, an activity is also not unique, but geo-feature--activity pairs (gf_i, $act_{i,j}$) are. Consequently, a place is cognitively, or spatially, including other features and the activities they afford.

Embedding this abstract model within the Resource Description Framework (RDF) [5], we obtain a data model represented by the vocabulary in Figure 3. In this diagram, rounded nodes represent entities consisting of several attributes; rectangular nodes denote literals, whose types are defined in other vocabularies in the Semantic Web[3]; edges report relations among nodes.

Figure 3: Vocabulary representing the data model for the abstract semantic model in
Figure 1.



This less-abstract model shows in more details the composition of entities. Places and geo-features are typically referred to by one or more names and may belong to a certain category or type (e.g., market, road). A geo-feature also has a geographic footprint that can be expressed, for example, as well-known text (geosparql:wktLiteral). Activities afforded by a geo-feature are simply represented by a pair of strings, denoting the verb and the context of the activity.

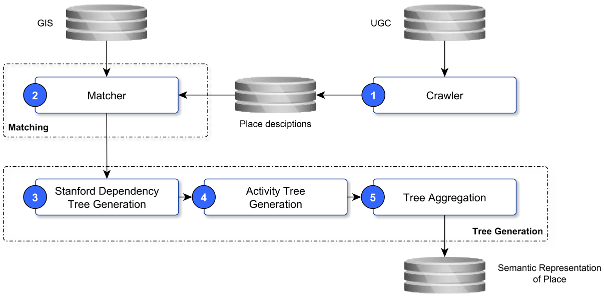
4 A Workflow to Derive Semantic Place Representations from Unstructured Text

As of today, information about the central part of the model in Figure 3 (i.e., extent, type, and name of geographic features) can be easily retrieved from many open and private sources. It remains still an open question, though, how to retrieve information about the rest of the model. Namely, we should find an answer to the following questions. Which geo-features constitute a place? Which activities these features afford?

We present an approach that uses Natural Language Processing (NLP) tools to extract this information from User Generated Content (UGC). More specifically, we present a workflow that uses the Stanford Dependency Parser [17] to process textual descriptions of places available on the Web (e.g., place reviews, touristic guides, travel logs, geo-blogs).

[3] xsd: http://www.w3.org/2001/XMLSchema#
geosparql: http://www.opengis.net/ont/geosparql#

Figure 4: Processing workflow to derive semantic place representations from unstructured text.



The workflow (see Figure 4) consists of 5 main steps:

1. A web crawler automatically collects place descriptions in the form of unstructured natural language texts referring to a given cognitive region—e.g., the *Historic Centre of Vienna*.
2. A spatial dataset of the area of interest is used as a knowledge base to match names and categories of geo-features against the textual descriptions. This step allows for detecting which geo-features constitute the place of interest.
3. Textual descriptions containing references to geo-features are processed with the Stanford Dependency Parser [17] to obtain a dependency tree—i.e., syntactical classes of the terms occurring in the text and a structured representation of the grammatical relations among them.
4. The resulting dependency tree is parsed to detect verb-context pairs making up activities that can be performed at or in the proximity of a given location. This step produces the activity sub-trees to be attached to a geo-feature node.
5. Activity sub-trees are aggregated by geo-feature to obtain the overall semantic representation of the place.

Step 1 (data crawling) can be performed separately and yields one of the two input data sources: a set *TD* of textual descriptions of a given place *pl*. The other data source is the set *GF* of geo-features in the area of interest. Step 2 (geo-matching) is assumed to be realized by the function *s.refers(gf)* appearing at line 3 of Algorithm 2 that detects whether a sentence *s* refers a geo-feature *gf* (either by name or by category). An implementation of this function based on regular expressions is provided in [12].
In the remainder of this section we present an algorithmic realization of the core part of the workflow (steps 3, 4, and 5).

The main function is reported in Algorithm 1. Given a place *pl*, a set *TD* of textual descriptions of *pl*, and a spatial dataset *GF*, the function GENERATEPLACETREE[4] produces a semantic representation t_{pl} of *pl* according to the model described in the previous section. As an example, assume that *pl* is the ”*Historic Centre of Vienna*”, *GF* is the OpenStreeMap dataset of the city of Vienna, and that *TD* comprises the text “*I even*

[4] We call the resulting representation a semantic tree, rather than graph, as the model describes a hierarchical structure without loops.

enjoyed walking down the beautiful Kärtnerstrasse admiring many nice, original shops”. The dependency tree of the latter is depicted in Figure 5, while the partial semantic representation derived from it is shown in Figure 6. As a first step (line 2) the algorithm creates the root node of the place tree (see Figure 6).

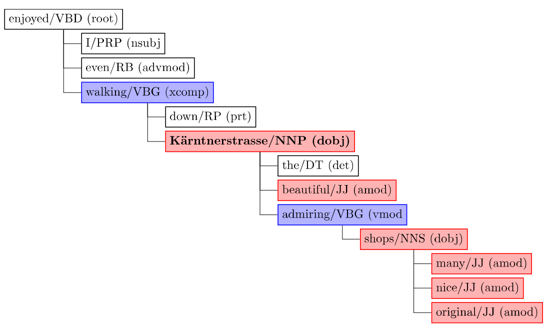
Algorithm 1: Given a place name *pl*, a set *TD* of textual descriptions referring to *pl*, and a geo dataset *GF*, produces a semantic representation of the place according to the model in Figure 3.

Input
pl = *a place/cognitive region*,
$GF = \{gf \mid gf$ *is a geofeature,*$\}$
$TD = \{td \mid td$ *is a textual description of pl*$\}$
Output
t_{pl} = *semantic tree description of pl*

1: **function** GENERATEPLACETREE
2: $t_{pl} \leftarrow initializePlaceTree(pl)$
3: $S \leftarrow getSentences(TD)$
4: **for all** $(gf, s) \in GF \times S$ **do**
5: $t_{gf} \leftarrow$ GENERATEGEOSUBTREE(gf, s)
6: **if** $t_{gf} \neq \emptyset \wedge t_{gf}.hasActivities()$ **then**
7: $t_{pl}.append(t_{gf})$
8: **return** t_{pl}

Subsequently (line 3), the textual descriptions are split into single sentences. This is done to avoid associating, later on, activities found in a sentence to the geo-feature(s) referred in another sentence of the description. For each geo--feature-sentence pair *(gf, s)* the algorithm calls (line 5) the function GENERATEGEOSUBTREE that is in charge of producing a so-called geo-sub-tree. By geo-sub-tree we intend the part of the semantic representation starting at the geo-feature node and proceeding all the way down to the verbs and contexts making up the activities (cf. Figure 3). If such a sub-tree is not empty and it includes at least one activity, it is appended to the place node t_{pl} (line 7).

Figure 5: Dependency tree obtained through the Stanford Dependency Parser [17] for the sentence “I even enjoyed walking down the beautiful Kärtnerstrasse admiring many nice, original shops”. The node labelled with bold text is the only geo-feature. Blue and red nodes indicate verbs and contexts of activities, respectively.



Algorithm 2: Given a geo-feature *gf* and a sentence *s* produces a semantic representation t_{gf} of *gf* according to the model in Figure 3.

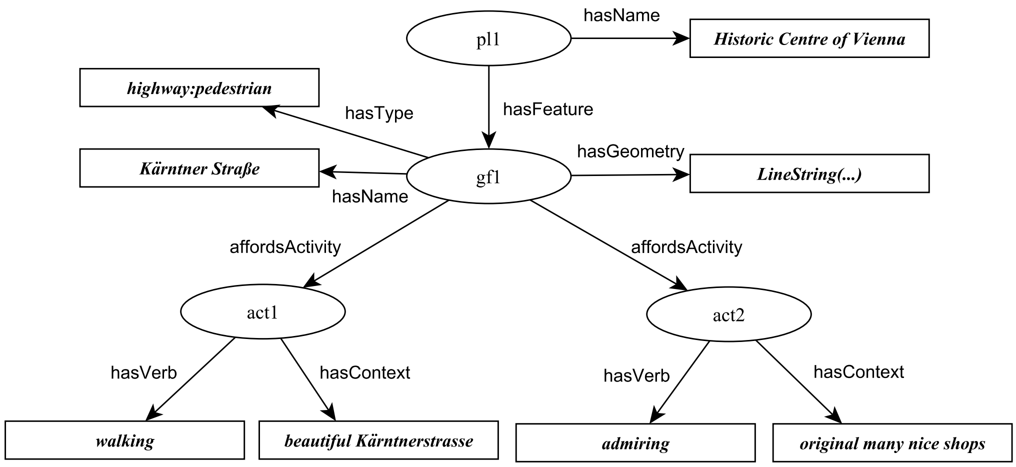
Input
gf = *a geofeature*,
s = *a sentence*
Output
t_{gf} = *semantic subtree description of a geo-feature*

1: **function** GENERATEGEOSUBTREE
2: $t_{gf} \leftarrow \emptyset$
3: **if** $s.refers(gf)$ **then**
4: $D \leftarrow getDependencyTrees(s)$
5: **for all** $d \in D$ **do**
6: $T_{act} \leftarrow$ GENERATEACTIVITYSUBTREES(d)
7: **if** $T_{act} \neq \emptyset$ **then**
8: $t_{gf} \leftarrow initializeGeoTree(gf)$
9: $t_{gf}.appendAll(T_{act})$
10: **return** t_{gf}

GENERATEGEOSUBTREE is described in Algorithm 2. An empty geo-sub-tree t_{gf} is initialized (line 2) which is actually built (line 8) only if the following two conditions hold true. (*i*) The input sentence *s* refers the input geo-feature *gf* (line 3). (*ii*) It is possible to associate at least an activity to *gf* (line 7). If the first condition is satisfied, a set *D* of dependency trees is generated from the given sentence (line 6). The function *getDependencyTrees(s)* resorts to the Stanford Dependency Parser [17] to generate a dependency tree for each independent clause of the sentence *s* (step 3 of the workflow in Figure 4). For the running example we have only one sentence and one independent clause, so only the tree in Figure 5 is generated where the node labelled with bold text indicates the only geo-feature for which a name match was found in the spatial dataset *GF*. Each such dependency tree (*d*) is given in input to the function GENERATEACTIVITYSUBTREE that is in charge of mapping from *d* onto the part of the semantic representation rooted at the activity node (cf. Figure 3). This function returns a (possibly empty) set T_{act} of activity-sub-trees: one for each verb-context pair found in the dependency tree. If the returned set is not empty, the geo-sub-tree is finally initialized (line 8) and the activity-sub-trees are appended to the geo-feature node t_{gf}. Note that the given algorithmic realization assumes data to be represented as Resource Description Framework (RDF) triples and persisted in a triple store. This means that a geo-feature *gf* corresponds to a uniquely identified node in the triple store. The function *initializeGeoTree(gf)* creates a new node only if *gf* is not already stored, otherwise it retrieves it from the triple store. This implements step 5 (aggregation) of the workflow in Figure 4.

GENERATEACTIVITYSUBTREE is described in Algorithm 3. The dependency tree under consideration possibly contains several verbs (e.g., enjoyed, walking, admiring), but we are only interested at those that also have a refining context. To achieve this we suggest using the grammatical relation *dobj* (direct object) that leads to the part of the sentence recognized as the (accusative) object of a verbal predicate [18]. The dependency tree in Figure 5 contains two such objects: *Kärtnarstrasse* and *shops*. For each such direct object *o* an activity-sub-tree t_{act} is initialized (line 5). The functions

Figure 6: (Partial) semantic representation (according to the model in Figure 3) of the Historic Centre of Vienna as derived by the description of Figure 5.



getVerb(o) and *getContext(o)* (line 6) start from the node *o* of the dependency tree and traverse it to retrieve the verb *vb* and the context *ctx*. By construction, the verb *vb* is always the nearest verbal (i.e. tagged VB) ancestor of *o* in the dependency tree. In our example, we get *walking* for the object *Kärtnerstrasse* and *admiring* for the object *shops*. We assume that the context consists of the object *o* itself plus all related adjectives (i.e. tagged JJ). By construction, these are located in the tree branch rooted at *vb* and going trough *o*. In the example we have *beautiful Kärtnerstrasse* and *original many nice shops*. Verb and context are appended (lines 7-8) to the root t_{act} of the activity-sub-tree, which is finally added to the return set T_{act} (line 9).

Algorithm 3: Given a dependency tree *d* produces a set T_{act} of activity sub-trees to be attached to a geo-feature sub-tree.

Input
$d = a\ dependency\ tree$
Output
$T_{act} = \{t_{act} \mid t_{act}\ is\ the\ semantic\ description\ of\ an\ activity\ act\}$

1: **function** GENERATEACTIVITYSUBTREE
2: $T_{act} \leftarrow \emptyset$
3: $O \leftarrow d.getDObjs()$
4: **for all** $o \in O$ **do**
5: $t_{act} \leftarrow initializeActivityTree()$
6: $vb, ctx \leftarrow getVerb(o), getContext(o)$
7: $t_{act}.appendVerb(vb)$
8: $t_{act}.appendContext(ctx)$
9: $T_{act} \leftarrow T_{act} \cup t_{act}$
10: **return** T_{act}

5 Discussion and Future Work

The examples reported in previous sections are real cases that we extracted from an implementation of the workflow described above. As geographic and User Generated Content data sources we used the Vienna extract of OpenStreetMap (as available on Metro Extracts[5]) and blog entries from TripAdvisor, respectively. The latter provides around 225 million crowd-sourced review pages for restaurants, hotels, and, more interestingly, what on the site is left unspecified as “places”. One such review page is titled “Historic Centre of Vienna”, which served as input for our web crawler.

One possible application for the presented model is that of enabling natural language query answering. In fact, a spatial question posed in natural language can be interpreted into the same (tree) semantic structure onto which we encoded unstructured place descriptions. Then, answering the query simply consists in matching two graphs.

The presented approach is part of on-going work but we believe it provides a good foundation for future improvements. While first experiments were performed on English texts, we plan for multilingualism [16] in future work. Two major simplifications were the neglect of (*i*) spatial relations and (*ii*) negations. The former can strongly modify the spatial location where an activity can be performed (e.g., many inexpensive shops can be found *outside* the city centre)—although this is supposedly rare in reviews that describe a place as they typically describe what one can do at a location, rather than away from it. One approach to tackle this challenge is to resort to ontologies of spatial relations, as provided, for example, by qualitative spatial calculi [2]. The case of negations is more concerned with Natural Language Processing and can be addressed with techniques used in sentiment analysis.

Another aspect that in the current model is disregarded concerns the semantic similarity of verbs, contexts, and activities. For example, the activity of “seeing historical landmarks” is obviously similar (if not even equivalent) to “admiring old monuments”. To address this aspect semantic

[5] https://mapzen.com/metro-extracts/

similarity of terms should be accounted for. An approach is to resort to a synonym structure as provided, e.g., by WordNet[6].

Acknowledgements

This research was partially funded by the Vienna University of Technology through the Doctoral College Environmental Informatics.

References

[1] Alazzawi, A.N., Abdelmoty, A.I. and Jones, C.B. 2012. What can I do there? Towards the automatic discovery of place-related services and activities. *International Journal of Geographical Information Science*. 26, 2 (2012), 345–364.

[2] Bateman, J.A., Hois, J., Ross, R. and Tenbrink, T. 2010. A linguistic ontology of space for natural language processing. *Artificial Intelligence*. 174, 14 (2010), 1027–1071.

[3] Battle, R. and Kolas, D. 2011. Geosparql: enabling a geospatial semantic web. *Semantic Web Journal*. 3, 4 (2011), 355–370.

[4] Berners-Lee, T., Hendler, J., Lassila, O. and others 2001. The semantic web. *Scientific american*. 284, 5 (2001), 28–37.

[5] Consortium, W.W.W. and others 2014. RDF 1.1 Concepts and Abstract Syntax. (2014).

[6] Egenhofer, M. and Mark, D. 1995. Naive Geography. *Spatial Information Theory A Theoretical Basis for GIS*. A. Frank and W. Kuhn, eds. Springer Berlin Heidelberg. 1–15.

[7] Fogliaroni, P. and Hobel, H. Implementing Naıve Geography via Qualitative Spatial Relation Queries.

[8] Gibson, J.J. 1979. *The ecological approach to visual perception*. Houghton Mifflin.

[9] Gibson, J.J. 1977. The theory of affordances. *Perceiving, Acting, and Knowing: Toward and Ecological Psychology*. R. Shaw and J. Bransford, eds. Erlbaum. 62–82.

[10] Goodchild, M.F. 2011. Formalizing place in geographic information systems. *Communities, Neighborhoods, and Health*. Springer. 21–33.

[11] Hahn, J., Fogliaroni, P., Frank, A.U. and Navratil, G. 2016. A Computational Model for Context and Spatial Concepts. *AGILE 2016*. Springer.

[12] Hobel, H., Fogliaroni, P. and Frank, A.U. 2016. Deriving the Geographic Footprint of Cognitive Regions. *AGILE 2016*. Springer.

[13] Khan, A., Vasardani, M. and Winter, S. 2013. Extracting spatial information from place descriptions. *Proceedings of the First ACM SIGSPATIAL International Workshop on Computational Models of Place* (2013), 62–69.

[14] Koubarakis, M. and Kyzirakos, K. 2010. Modeling and querying metadata in the semantic sensor web: The model stRDF and the query language stSPARQL. *The semantic web: research and applications*. Springer. 425–439.

[15] Kuhn, W., Kauppinen, T. and Janowicz, K. 2014. Linked data-A paradigm shift for geographic information science. *Geographic Information Science*. Springer. 173–186.

[16] Laurini, R. 2015. Geographic Ontologies, Gazetteers and Multilingualism. *Future Internet*. 7, 1 (2015), 1–23.

[17] Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J. and McClosky, D. 2014. The Stanford CoreNLP natural language processing toolkit. *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (2014), 55–60.

[18] De Marneffe, M.-C. and Manning, C.D. 2008. *Stanford typed dependencies manual*. Technical report, Stanford University.

[19] Montello, D.R. 2003. Regions in geography: Process and content. *Foundations of geographic information science*. (2003), 173–189.

[20] Al-Salman, R., Dylla, F. and Fogliaroni, P. 2012. Matching geo-spatial information by qualitative spatial relations. *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Crowdsourced and Volunteered Geographic Information* (2012), 38–45.

[21] Scheider, S. and Janowicz, K. 2014. Place Reference Systems: A Constructive Activity Model of Reference to Places. *Appl. Ontology*. 9, 2 (2014), 97–127.

[22] Stadler, C., Lehmann, J., Höffner, K. and Auer, S. 2012. Linkedgeodata: A core for a web of spatial open data. *Semantic Web*. 3, 4 (2012), 333–354.

[23] Vasardani, M., Winter, S. and Richter, K.-F. 2013. Locating place names from place descriptions. *International Journal of Geographical Information Science*. 27, 12 (2013), 2509–2532.

[24] Zhang, Y., Gao, Y., Xue, L., Shen, S. and Chen, K. 2008. A common sense geographic knowledge base for GIR. *Science in China Series E: Technological Sciences*. 51, 1 (2008), 26–37.

[6] http://wordnet.princeton.edu/