

Danish GIS resources

Finn Årup Nielsen

June 15, 2017

Abstract

A range of different Danish geographical information system (GIS) resources, datasets and tools are presented. The focus is on resources for use in automated computational systems and free resources that can be redistributed and used in commercial applications.

Contents

1 Datasets	2
2 DAWA	2
3 OpenStreetMap	3
4 Wikidata	3
4.1 Named entity recognition	3
4.2 Distance functions	4

1 Datasets

Styrelsen for Dataforsyning og Effektivisering makes several datasets available from their homepage at <https://download.kortforsyningen.dk/>. These are part of the free Danish geodata that was released in 2013 [1, 2].

For instance, *Stednavne* contains Danish placenames with coordinates in GML and other formats. The names in the *Stednavne* GML file may be read with the following Python program:

```
from lxml import etree
tree = etree.parse('KORT10.gml')
path = '//{http://schemas.kms.dk/kf_dwl/kort10/2014/11/10}Navn'
[elem.text for elem in tree.findall(path)]
```

The *KORT10.gml* file is included in the *DK_GML_UTM32-EUREF89.zip* file distributed by Kortforsyningen. It contains 139'164 names.

2 DAWA

Danmarks Adressers Web API (DAWA) developed by *Styrelsen for Dataforsyning og Effektivisering* and available from <https://dawa.aws.dk/> presents an API for searching for Danish addresses in a variety of ways.

One of there API will return information based on a address query. As an example consider the query “Rentemestervej 8 2400” formulated in Python:

```
>>> import requests
>>> data = requests.get('https://dawa.aws.dk/adresser',
                        params={'q': 'Rentemestervej_8_2400'}).json()
>>> entry = data[0]
>>> entry['adgangsadresse']['adgangspunkt']['koordinater']
[12.5354718548566, 55.7048195502325]
```

Here there request will return a list of relevant locations, where the first might be the one most relevant. For each entry returned there is a number of information associated beyond the geographical coordinate, e.g., the *politikreds*, municipality, street, etc.

Various DAWA repositories are available from the GitHub at <https://github.com/DanmarksAdresser/>.

Name	Description
Stednavne	Danish placenames with coordinates in GML and other formats available via Kortforsyningen
Wikidata dump	Wikidata as JSON, RDF or XML

Table 1: Datasets

3 OpenStreetMap

OpenStreetMap (OSM) is an open data world map. OpenStreetMap has extensive updated maps of Denmark with interactive route finding. It is available from <https://www.openstreetmap.org>. There are various ways to interact with OSM, e.g., by the [Overpass API](#).

LinkedGeoData at <http://linkedgeo.org> is an effort to export OpenStreetMap information to the Linked Open Data cloud. It has a SPARQL endpoint at <http://linkedgeo.org/sparql>.

4 Wikidata

Wikidata is a sister site to Wikipedia containing structured information [3].

As of June 2017, Wikidata has 43'702 items where the item has been (directly) associated with Denmark and has a geo-coordinate.¹ These items can be queried from the Wikidata Query Service at <https://query.wikidata.org> with the following SPARQL query:

```
SELECT DISTINCT ?place WHERE {
  ?place wdt:P17 wd:Q35 .
  ?place wdt:P625 ?geo .
}
```

The data comes with labels and variations (aliases).

Organizations may not be directly associated with a geo-coordinate but have an associated geo-coordinate as a qualifier to the head quarter property.

4.1 Named entity recognition

An application using Wikidata data is named entity extraction of geo-referenceable names in natural language texts. A prototype for this application is implemented in the `stednavn` Python module available from <https://github.com/fnielsen/stednavn>. An application of this module on the Danish sentence “Lyngby Hovedgade er i Kongens Lyngby ikke i København eller Ørum” from within Python3 looks like this:

```
>>> from stednavn import Stednavn
>>> stednavn = Stednavn()
>>> s = ('Lyngby_Hovedgade_er_i_Kongens_Lyngby_ikke_'
...     'i_København_eller_Ørum.')
>>> stednavn.extract_placenames_from_string(s)
['Lyngby_Hovedgade', 'Kongens_Lyngby', 'København', 'Ørum']
```

¹The number of Danish Wikidata items with a geo-coordinate can be counted with <http://tinyurl.com/y7k3wggg> that uses the following SPARQL query

```
SELECT (COUNT(DISTINCT ?place) AS ?count) WHERE {
  ?place wdt:P17 wd:Q35 ; wdt:P625 ?geo .
}
```

Here the module extracts the four different named entities to a list of strings. The module may also be used as a script. The following lines download a historical novel, *Bent Bille*, from Runeberg as a text file and then extract geo-locatable named entities:

```
curl "http://runeberg.org/download.pl?mode=ocrtext&work=bentbille" \  
> bentbille.txt  
python -m stednavn bentbille.txt
```

The last command extracts currently 333 words or phrases on the command-line from the 57'898 words document in a matter of a few seconds. The first few lines of the result are listed here:

```
Sjælland  
Kloster  
Paris  
Radsted  
Lolland  
Søholm  
Paris  
Borup  
København  
...
```

There are various problems with the this simple approach.

1. Different entities may have similar names, e.g., “Lyngby” may be one of several separate places in Denmark. Currently there are no way of automatically selecting between the various versions.
2. Some named entities (proper nouns) resemble common nouns, e.g., “Bispen” is the Danish noun for “The bishop”, but is also the name of the cultural institution in Haderslev. “Kloster” in the above example with *Bent Bille* is likely also a similar error. The `stednavn` Python module maintains a stopword list (`stopwords-da.txt`) of currently 72 words for partially handling these cases. As the case with “Kloster” shows, this list is not complete.
3. The number of places are limited, e.g., only a minority of Danish street names is in Wikidata.

On the positive side, Wikidata records not only inheritly geographical items (towns, streets, etc.) but also items such as companies, events, sculptures and a range of other types of items that can be associated with a geo-coordinate. Such information cannot directly be found with DAWA.

4.2 Distance functions

The Python module `fromtodk` can return coordinates from Wikidata items and compute the distance between two Wikidata items via geopy’s vincenty function. `fromtodk` is available from <https://github.com/fnielsen/fromtodk> and a web application is running as a prototype from <https://fromtodk.herokuapp.com/>. It can compute the distance

between, e.g., the university department *DTU Compute* and the sculpture *Storkespringvandet*. With the Heroku-based fromtodk webservice the URL is:

<https://fromtodk.herokuapp.com/?f=DTU+Compute&t=Storkespringvandet>

It presently reports 12.3 kilometers. The command-line version would look like this:

```
$ python -m fromtodk "DTU□Compute" "Storkespringvandet"
12.3085761921
```

WDQS also enables geo-based distance computations. For instance, to compute the closed Wikidata-recorded items around (12.53547, 55.70482) (i.e., Rentemestervej 8) a SPARQL query may be formulated as

```
SELECT ?location ?locationLabel WHERE {
  BIND("Point(12.53547□55.70482)"^^geo:wktLiteral AS ?query)
  SERVICE wikibase:around {
    ?location wdt:P625 ?geo .
    bd:serviceParam wikibase:center ?query .
    bd:serviceParam wikibase:radius "1" .
  }
  SERVICE wikibase:label { bd:serviceParam wikibase:language "da,en" . }
  BIND(geof:distance(?query, ?geo) as ?dist)
}
ORDER BY ?dist
LIMIT 5
```

It will return the items for Geodatastyrelsen, Bygmestervej, Nørrebro Bycenter, Bispebjerg Kollegiet and Lygten.

Acknowledgement

This work was supported by the [Danish Innovation Foundation](#) (Innovationsfonden) through the project Danish Center for Big Data Analytics and Innovation (DABAI).

References

- [1] Bente Steffensen, Tina Svan Colding, and Anne Marie Carstens. [Frie geodata har fået antallet af brugere på Kortforsyningen.dk til at eksplodere](#). *Geoforum Perspektiv*, 13:49–53, December 2014.
- [2] Bente Steffensen, Tina Svan Colding, and Anne Marie Carstens. [Free Geodata Have Made the Number of Users of the Map Supply Website Explode](#). *Geoforum Perspektiv*, 13:49–53, December 2014.
- [3] Denny Vrandečić and Markus Krötzsch. [Wikidata: a free collaborative knowledge-base](#). *Communications of the ACM*, 57:78–85, October 2014.

Index

DAWA, [2](#)

GML, [2](#)

OpenStreetMap, [2](#), [3](#)

SPARQL, [3](#)

Stednavne, [2](#)

Wikidata, [2](#), [3](#)