

## Geo::Parser::Text

Geo Parsing and Geo coding locations from Text.

Ervin Ruci - geocode.xyz - YAPC::NA 2016

## About this talk

- 1 I gave the first version of this talk at FOSDEM 2016 (Geospatial Devroom)
- 2 Some members of the Audience expressed surprise at my choice of Perl for such an application (?!)
- 3 So, here is an expanded version aimed at a Perl friendly audience
- 4 This talk (and various test data) are hosted at <https://github.com/eruci/openaddresses/tree/master/test>
- 5 The main application is available as a server image on AWS and <https://metacpan.org/pod/Geo::Parser::Text>

## Text => Geo Parsed Locations in Text

This and That and the Other street in Porters Lake Nova Scotia

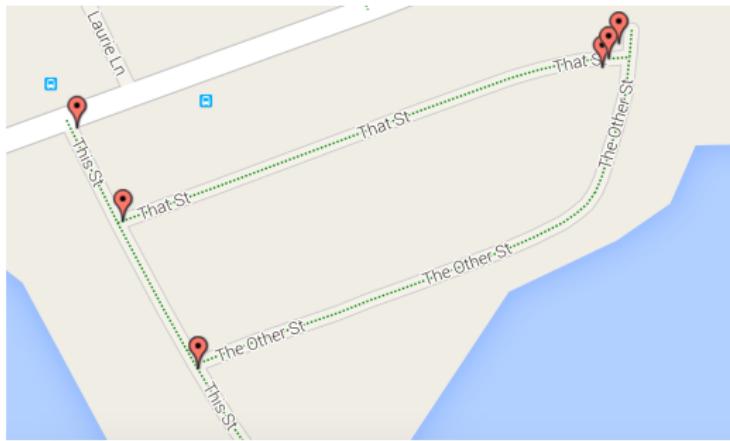
Q. How many locations are in this text?

# The Problem: Text => Geo Locations in Text

This and That and the Other street in Porters Lake Nova Scotia

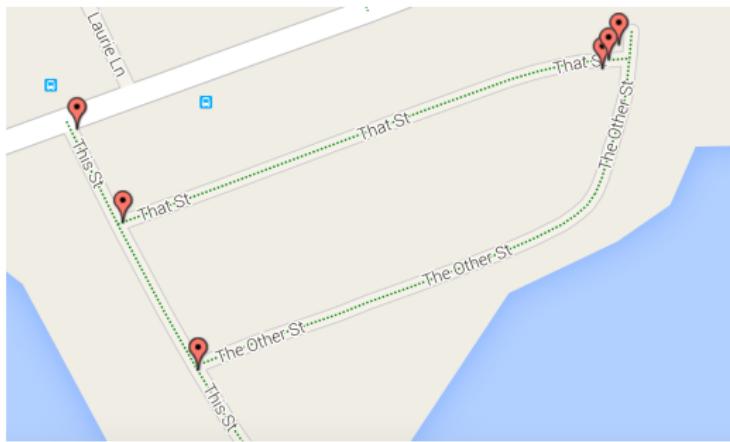
Q. How many locations are in this text?

A. 6. <http://geocoder.ca>



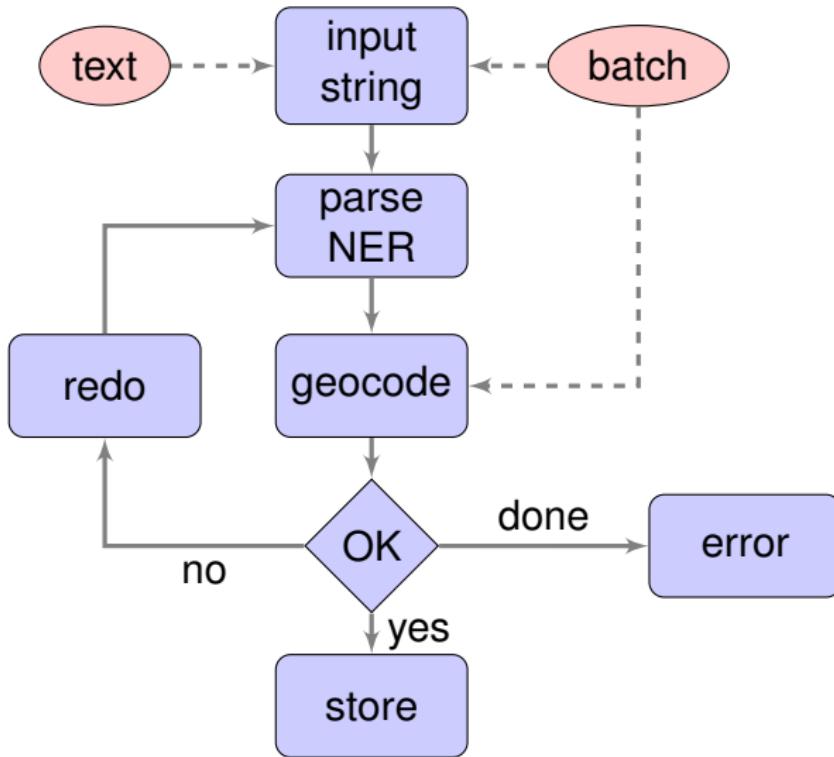
# The Problem: Text => Geo Locations in Text

- 1 THIS ST AND THAT ST, PORTERS LAKE, NS
- 2 THIS ST, PORTERS LAKE, NS
- 3 THAT ST, PORTERS LAKE, NS
- 4 THE OTHER ST AND THAT ST, PORTERS LAKE, NS
- 5 THE OTHER ST AND THIS ST, PORTERS LAKE, NS
- 6 THE OTHER ST, PORTERS LAKE, NS



Reqs: Identify addresses, intersections, city names, province/state.

# The Solution: Geoparsing and Geocoding



# Geocoding Spans Many Fields

- 1 linguistics (matching/translating across different languages)
  - 2 data processing (normalization, standardization and input)
  - 2.1 data structures (R-trees, KD-trees,...)
  - 3 natural language processing (parsing, named entity recognition)
  - 4 computational geometry (point in polygon)
  - 5 pattern recognition (fuzzy match)
  - 6 geography (dealing with projections)
  - 7 Ai (learning, hidden markov models)
  - ... and a few others (tokenization, data cleanup, UI..)
- AND** Testing, testing and more testing

## And there are Many Geocoders.

And Many more are being built, Plus a few I've tested:

- 1 Google Geocoder (Coverage: 99%, Accurate 93%) (Canada)
- 2 HERE.com (Coverage 98%, Accurate 92%) (Canada)
- 3 Nominatim (Coverage 80%, Accurate 57%) (Canada)
- 4 Geocoder.ca (Coverage 99%, Accurate 94%) (Canada)
- 5 Geocode.xyz (Coverage 80%, Accurate 58%) (Spain)
- 6 Mapzen.com (Coverage 86%, Accurate 80%) (Spain)

Download test data and results here: <https://github.com/eruci/openaddresses/tree/master/test>

Perceptions on quality in open source geocoders are mostly negative...

# Why create a new Geocoder/Geoparser?

No Geocoder does it all. Google Geocoder (presumably the most complete in the bunch) does not (most importantly)

- 1 Provide 100% coverage (open problem)
- 2 Provide 100% accuracy (open problem too) Also...
  - a Geocode parcel data (avail as opendata in Canada and USA)
  - b Extract location data from text (geoparsing)
  - c Do address parsing and standardization (incl postal codes, in Canada it only provides 3 letter FSA)
  - d Provide unlimited API access (throttling/rate limiting/geo blocking/ geo data may not be retained etc ) ...

# The key ingredients of the solution

- 1 DATA
- 2 A good parser

# Parsing

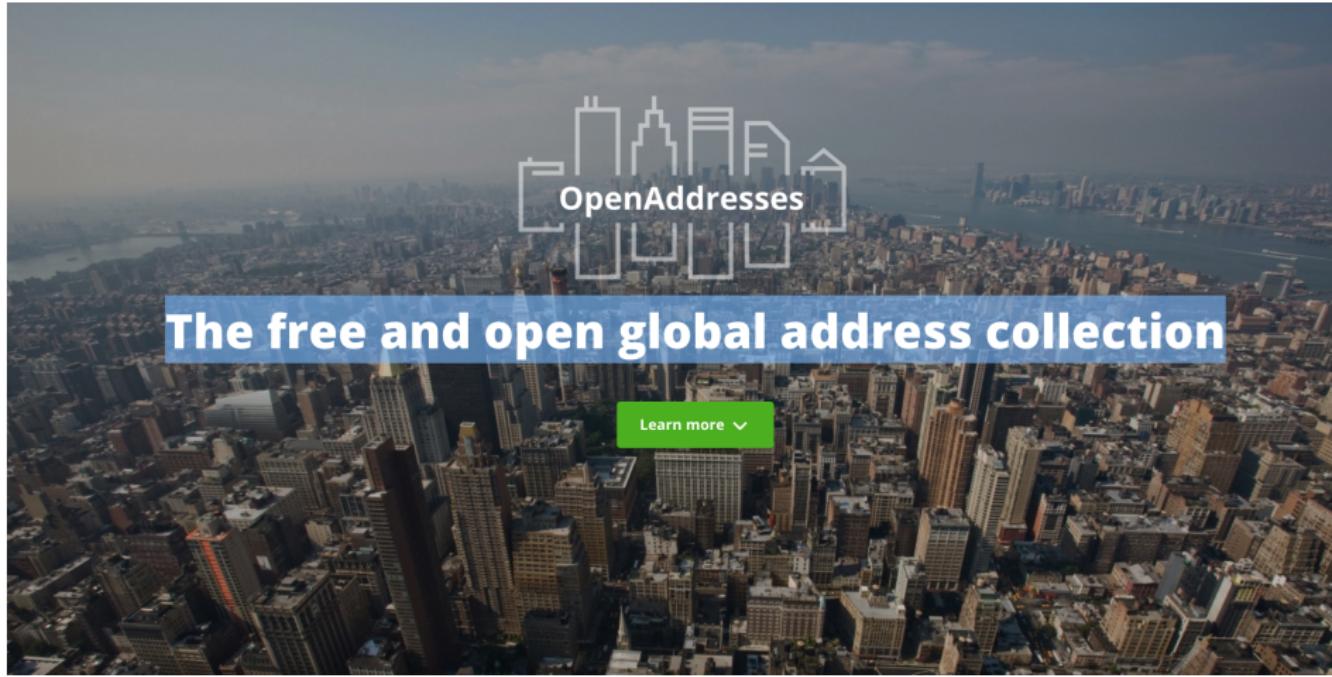
- 1 Fuzzy vs Exact (correct spelling errors)
- 2 Partial vs Complete (fill in missing location entities)
- Quick demo: <http://geocoder.ca/textscan>

## Accuracy and Coverage Differ Because

- a Geocoding/Geoparsing is an imprecise process and various Geocoders fail in various ways.
- b Ambiguities, incomplete data, incorrect data, software bugs,

(Open) Data - Openaddresses.io, openstreetmap.org,

...



over 248,548,165 addresses (was half that only 6 months ago)

# Even Google Maps (presumably the best) Fails!

- ✖ Even in well mapped big cities.

# Even when you Google wrong location you get:



wrong location |



wrong location

wrong location **on google maps**

wrong location **on google**

wrong location **on facebook**

wrong location **tinder**

wrong location **on iphone**

wrong location **on phone**

wrong location **android**

wrong location **on traffic ticket**

wrong location **on parking ticket**



: Wrong Location Google Maps!.

# Geoparsing and Geocoding in unstructured text

Text (from wikipedia entries to microblog posts) => Geocoded Locations.

- 1 extraction
- 2 disambiguation
- 3 geocoding

Demo: <http://geocode.xyz>

## Geo::Parser::Text

```
use Geo::Parser::Text;
my $g = Geo::Parser::Text->new('http://geocoder.ca');
my $text = "The Downtown Orlando Information kiosk is somewhere on
Orange Ave";
my $ref = $g->geocode(scantext=>$text,region=>'FL');
```

## Geo::Parser::Text

```
perl t.pl
$VAR1 = { 'match' => {
'staddress' => 'Orange Ave N',
'stnumber' => {},
'longt' => '-81.3718620000',
'prov' => 'FL',
'city' => 'Orlando',
'confidence' => '0.5',
'latt' => '28.5795573000'
},
};
```

# Putting the World's open information on a Map

Sample applications:

- 1 openwikimap.org (Wikipedia => Openstreetmap)
- 2 wherewords.xyz (Common Crawl => word frequencies by location; a more intuitive version of what3words)
- 3 WhereNewsNow.xyz (Current News on a Map, scan newsfeeds, twitter/social media feeds for locations map them)

# In case internet does not work for the demo



Toronto Police OPS @TPSOoperations · Jan 26

Robbery investigation on @GOtransit. Suspect pulled brake, hopped off train at Birchmount Rd/Raleigh Ave on LakeShore line. ^vk



17

8

•••

: [Twitter Feed Demo geocoder.ca]

40 locations within a 3484.180 km radius, found in this text:

Toronto Police OPS Verified account @TPSOoperations · Jan 26

Robbery investigation on @GOtransit. Suspect pulled brake, hopped off train at Birchmount Rd/Raleigh Ave on LakeShore line. ^vk

Reprocess text and Download results

On a Map



#### Match Location

1	Raleigh Ave And Birchmount Rd, Toronto, On	Confidence Score: 1
2	Raleigh, Nc	Confidence Score: 0.8
3	Toronto, On	Confidence Score: 0.8
4	Raleigh, Ms	Confidence Score: 0.8
5	Raleigh, Fl	Confidence Score: 0.8
6	Toronto, Pe	Confidence Score: 0.8
7	Toronto, Ks	Confidence Score: 0.8
8	Toronto, In	Confidence Score: 0.8
9	Raleigh, Wv	Confidence Score: 0.8
10	Toronto, Tx	Confidence Score: 0.8

# In case internet does not work for the demo

Geocoder.ca Services | Products | Solutions Terms Login Create Account API Contact

Found 5 locations in this line:

6 locations within a 321.011 km radius of Birchmount Rd/Raleigh Ave, ON

Toronto Police OPS Verified a Robbery investigation on @G

Reprocess text and Data

Suspect pulled brake, hopped off train at Birchmount<sup>1</sup> Rd<sup>1</sup>/Raleigh<sup>1</sup> Ave<sup>1</sup> on LakeShore line

1. RALEIGH AVE AND BIRCHMOUNT RD, TORONTO, ON (Confidence: 1)  
2. LAKESHORE AVE, TORONTO, ON (Confidence: 0.8)  
3. RALEIGH AVE, TORONTO, ON (Confidence: 0.6)  
4. BIRCHMOUNT RD, TORONTO, ON (Confidence: 0.5)  
5. LakeShore, ON (Confidence: 0.5)

A map of the Greater Toronto Area (GTA) in Ontario, Canada. The map shows major cities like Toronto, Mississauga, Brampton, Vaughan, Guelph, Waterloo, Kitchener, and Hamilton. Specific locations marked with blue dots include Birchmount Rd, Raleigh Ave, and LakeShore Ave. The map also includes labels for Walkerton, Minto, Whitby, Oshawa, Cobourg, Stratford, and Cob. A legend on the right side lists five geoparsed locations with their confidence levels.

1	Raleigh Ave And Birchmount Rd, Toronto, On	Confidence: 1
2	Lakeshore Ave, Toronto, On	Confidence: 0.8
3	Toronto, On	Confidence: 0.6
4	Raleigh Ave, Toronto, On	Confidence: 0.5
5	Birchmount Rd, Toronto, On	Confidence: 0.5

: [Geoparsed]

# In case internet does not work for the demo

**GeoCode.xyz** API

2 locations within a 1.084 km radius, found in this text:

The most important museums of Amsterdam are located on the Museumplein (Museum Square), located at the southwestern side of the Rijksmuseum.

Reprocess text and Download results On a Map



Match	Location	Confidence Score
1	Amsterdam, NL	0.7
2	Museumplein, Amsterdam, NL	0.3

Click a marker for more information.

: [Demo geocode.xyz]

# In case internet does not work for the demo

GeoCode.xyz

Found 2 locations in this line:

The most important museums of Amsterdam<sup>2</sup> are located on the Museumplein<sup>2</sup> (Museum Square), located at the southwestern side of the Rijksmuseum

1. [Amsterdam, NL](#) (Confidence: 0.7)  
2. [MUSEUMPLEIN, AMSTERDAM, NL](#) (Confidence: 0.3)

2 locations within a 1.084 km radius:  
The most important museum in Amsterdam is the Rijksmuseum.

Reprocess text and Data

A map of Amsterdam with the city center highlighted in red. The 'Museumplein' is marked with a blue dot in the southern part of the city. The map also shows the 'GRACHTENGORDEL' (canal belt) and various neighborhoods like 'AMSTERDAM-WEST', 'OVERTOMSE VELD', 'WEESPERBUURT EN PLANTAGE', 'INDISCHE BUURT', and 'DOSTERPARKBUURT'. A legend on the right indicates 'Match' and 'Location' for the two found matches.

Match	Location	Confidence Score
1	Amsterdam, NL	0.7
2	Museumplein, Amsterdam, NL	0.3

Click a marker for more information.

: [Demo geocode.xyz]

# In case internet does not work for the demo

GeoCode.xyz

API

7 locations within a 1.803 km radius, found in this text:

Bruxelles/Brussel - Brussels encompasses many charming and beautiful attractions, with deeply ornate buildings on the Grand Place/Grote Markt, and a fish-and-crustacean overdose of St. Catherine's Square (Place St-Catherine/Sint-Katelijneplein). Stroll along, (and stop in for a drink)

Reprocess text and Download results

On a Map



Match	Location
1	Antoine Dansaertstraat, Brussels, BE Confidence Score: 0.8
2	Brussels, BE Confidence Score: 0.7
3	Antoine Dansaert Rue, Brussels, BE Confidence Score: 0.7
4	Sint-Goriksplein, Brussels, BE Confidence Score: 0.6
5	Sint-Katelijneplein, Brussels, BE Confidence Score: 0.4
6	Grote Markt, Brussels, BE Confidence Score: 0.2
7	Grand Place, Brussels, BE Confidence Score: 0.2

: [Demo geocode.xyz]

# In case internet does not work for the demo

GeoCode.xyz API

Found 7 locations in this line:

7 locations within a 1.803 km

Bruxelles/Brussel - Brussels Markt, and a fish-and-crustacean overdose of St. Catherine's Square (Place St-Catherine/Sint-Katelijneplein), Stroll along, (and stop in for a drink) at one of the many bars on Place St-GÃ©ry/Sint-Goriksplein, or max out your credit card on the trendy Rue Antoine<sup>3</sup> Dansaert<sup>3</sup>/Antoine<sup>3</sup> Dansaertstraat

Reprocess text and D

ANTOINE DANSAERTSTRAAT, BRUSSELS, BE (Confidence: 0.8)

2. Brussels, BE (Confidence: 0.7)

3. ANTOINE DANSAERT RUE, BRUSSELS, BE (Confidence: 0.7)

4. SINT-GORIKSPLAAT, BRUSSELS, BE (Confidence: 0.6)

5. SINT-KATELIJNEPLEIN, BRUSSELS, BE (Confidence: 0.4)

6. GROTE MARKT, BRUSSELS, BE (Confidence: 0.2)

7. GRAND PLACE, BRUSSELS, BE (Confidence: 0.2)

The map shows the central area of Brussels with several location markers. One marker is explicitly labeled 'Antoine Dansaertstraat' with a pin icon. Other markers are scattered across the map, representing the other locations found in the search results.

1. ANTOINE DANSAERTSTRAAT, BRUSSELS, BE  
2. Brussels, BE  
3. ANTOINE DANSAERT RUE, BRUSSELS, BE  
4. SINT-GORIKSPLAAT, BRUSSELS, BE  
5. SINT-KATELIJNEPLEIN, BRUSSELS, BE  
6. GROTE MARKT, BRUSSELS, BE  
7. GRAND PLACE, BRUSSELS, BE

: [Demo geocode.xyz]

# Coding a Geocoder/Geoparser that does this is easy (in theory)

But.. making it recognize over 90% of input at over 90% accuracy requires at least these steps

- 1 importing and parsing country specific data from openaddresses.io (suffixes, prefixes, city names, numbering schemes)
- 2 cleaning up errors post import.
- 3 test and pick away at errors, one at a time

In theory, there is no difference between theory and practice. But, in practice, there is.

# Perl makes easy problems very easy

.. and hard ones, easy. (in both theory and practice)

- 1 No other language is better at slicing and dicing text.
- 2 CPAN has lots of valuable bits for any NLP puzzle such as this one
- 3 Do I need another reason to use perl?

## Go ahead and test it

No rate limit on: <http://geocode.xyz>

Let us know of bugs/feature suggestions

G  
e  
o  
c  
o  
d  
e  
r  
.c  
a

Contact: e: [eruci@geocoder.ca](mailto:eruci@geocoder.ca) twitter: [@geolytica](https://twitter.com/geolytica)

*PS. One more thing. The core module is 57355 lines of Perl code.*