



*Automatically Geotagging Unstructured Text
with Open Source Tools*

Charlie Greenbacker, Principal Data Scientist

Background



◆ About Me:

- ◆ Data Scientist
- ◆ Natural Language Processing
- ◆ Unstructured Data → Information

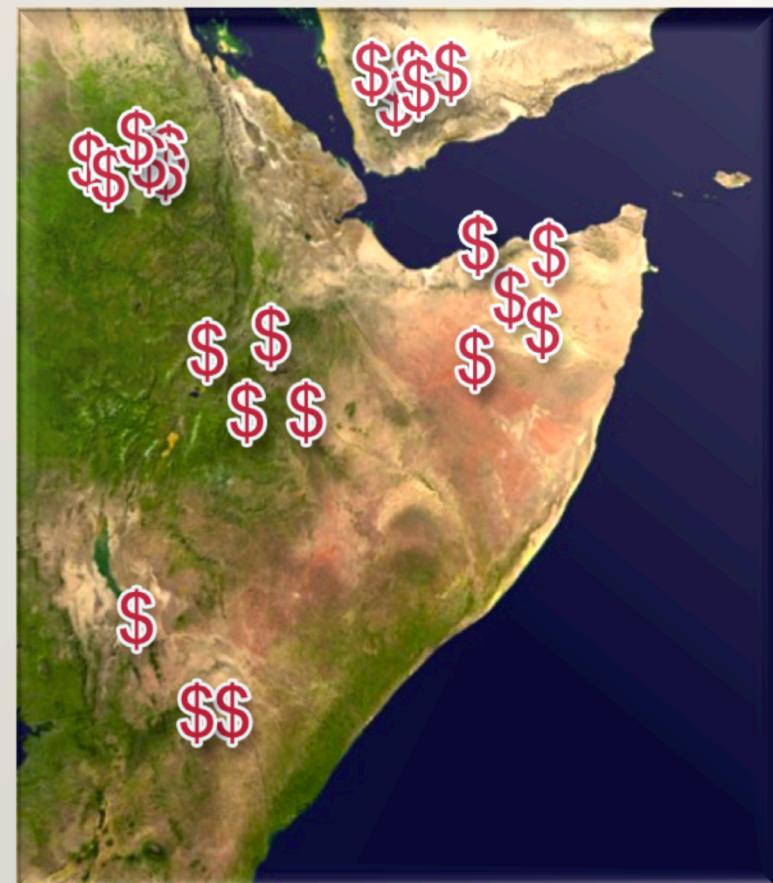
◆ Berico Technologies:

- ◆ Veteran-owned Small Business
- ◆ Open Source Software & Services
- ◆ Big Data Analytics in the Cloud
- ◆ Defense & Intel Community

Problem: Geotagging Unstructured Text

- ◆ Growing demand for geospatial analytics
- ◆ Most human knowledge is “trapped” in text
- ◆ Existing solutions are **expensive** & don’t scale

- ◆ Need an **open source** solution!



Solution: an Open Source Geoparser



- 1. Ingest unstructured text**
- 2. Extract place names**
 - ◆ Geo entity extraction
- 3. Disambiguate names**
 - ◆ Geo entity resolution
- 4. Enrich text w/ geo data**

Ingest Unstructured Text

Comment



photo: Flickr user NS Newsflash

Extract Place Names

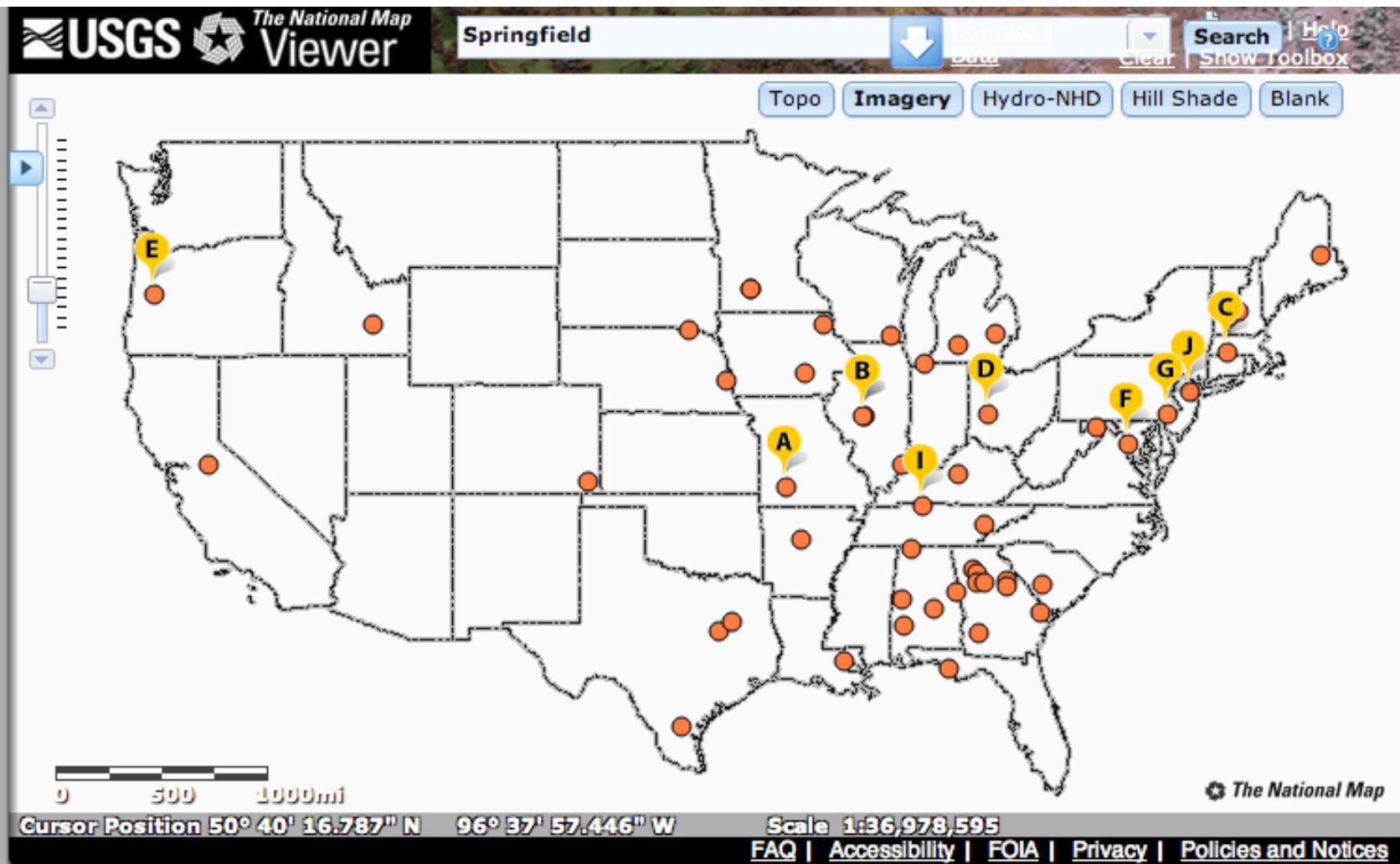
- ◆ Use existing open source tools for Named Entity Recognition:



Disambiguate Place Names



“The Springfield Problem”



Enrich Text with Geo Data

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	8504755	Kapicik	39.14902	46.0038	T	PK	AM				0		3447	Asia/Baku	3/21/13
2	2772493	Lieserhofen	46.83642	13.48838	P	PPL	AT		2		0		647	Europe/Vienna	3/21/13
3	145482	Kaputjugh Lerr	39.15898	46.00384	T	MT	AZ		0		0	3904	3576	Asia/Baku	3/21/13
4	146938	Yuxari Aza	38.92963	45.82517	P	PPL	AZ		35		0		740	Asia/Baku	3/21/13
5	146958	Yayci	38.9427	45.7319	P	PPL	AZ		35		0		707	Asia/Baku	3/21/13
6	147165	Surud	39.14618	45.79992	P	PPL	AZ		35		0		1336	Asia/Baku	3/21/13
7	147365	Ordubad Rayon	39.08333	45.91667	A	ADM2	AZ	35	147365	42638			1895	Asia/Baku	3/21/13
8	147638	Kyuznut	39.1288	45.53339	P	PPL	AZ		35		0		880	Asia/Baku	3/21/13
9	147703	Koshadiza	38.95361	45.83257	P	PPL	AZ		35		0		805	Asia/Baku	3/21/13
10	147761	Xokesin	39.17052	45.69667	P	PPL	AZ		35		0		1198	Asia/Baku	3/21/13
11	147810	Khanakakh	39.19211	45.70732	P	PPL	AZ		35		0		1262	Asia/Baku	3/21/13
12	147840	Kyarimkuli-Diza	39.01158	45.74214	P	PPL	AZ		35		0		856	Asia/Baku	3/21/13
13	147874	Karudzhikh	39.16402	45.99957	P	PPL	AZ		35		0		3700	Asia/Baku	3/21/13
14	147948	Kalantardiza	38.95116	45.82592	P	PPL	AZ		35		0		790	Asia/Baku	3/21/13
15	148045	Gilancay	38.91624	45.81617	H	STM	AZ	AZ	35		0		704	Asia/Baku	3/21/13
16	148132	Julfa Rayon	39.16667	45.66667	A	ADM2	AZ	35	148132	38554			1290	Asia/Baku	3/21/13
17	148133	Gueluestan	38.9834	45.5895	P	PPL	AZ	AZ	35		482		741	Asia/Baku	3/21/13
18	148153	Camaldin	39.09242	45.60123	P	PPL	AZ		35		0		959	Asia/Baku	3/21/13
19	148168	Dyuylun	38.95333	45.88667	P	PPL	AZ		35		0		910	Asia/Baku	3/21/13
20	148251	Culfa	38.9558	45.6308	P	PPLA2	AZ		35		10820		715	Asia/Baku	3/21/13
21	148370	Bashdiza	38.98492	45.82691	P	PPL	AZ		35		0		869	Asia/Baku	3/21/13
22	148399	Bakhrud	39.07559	45.86513	P	PPL	AZ		35		0		1371	Asia/Baku	3/21/13
23	148483	Aza	38.91972	45.82104	P	PPL	AZ		35		0		712	Asia/Baku	3/21/13
24	394446	Culfa Stansiyasi	38.95775	45.62313	S	RSTN	AZ		35		0		717	Asia/Baku	3/21/13
25	394448	Yayci Yolayricisi	38.93272	45.74729	S	RSD	AZ		35		0		701	Asia/Baku	3/21/13

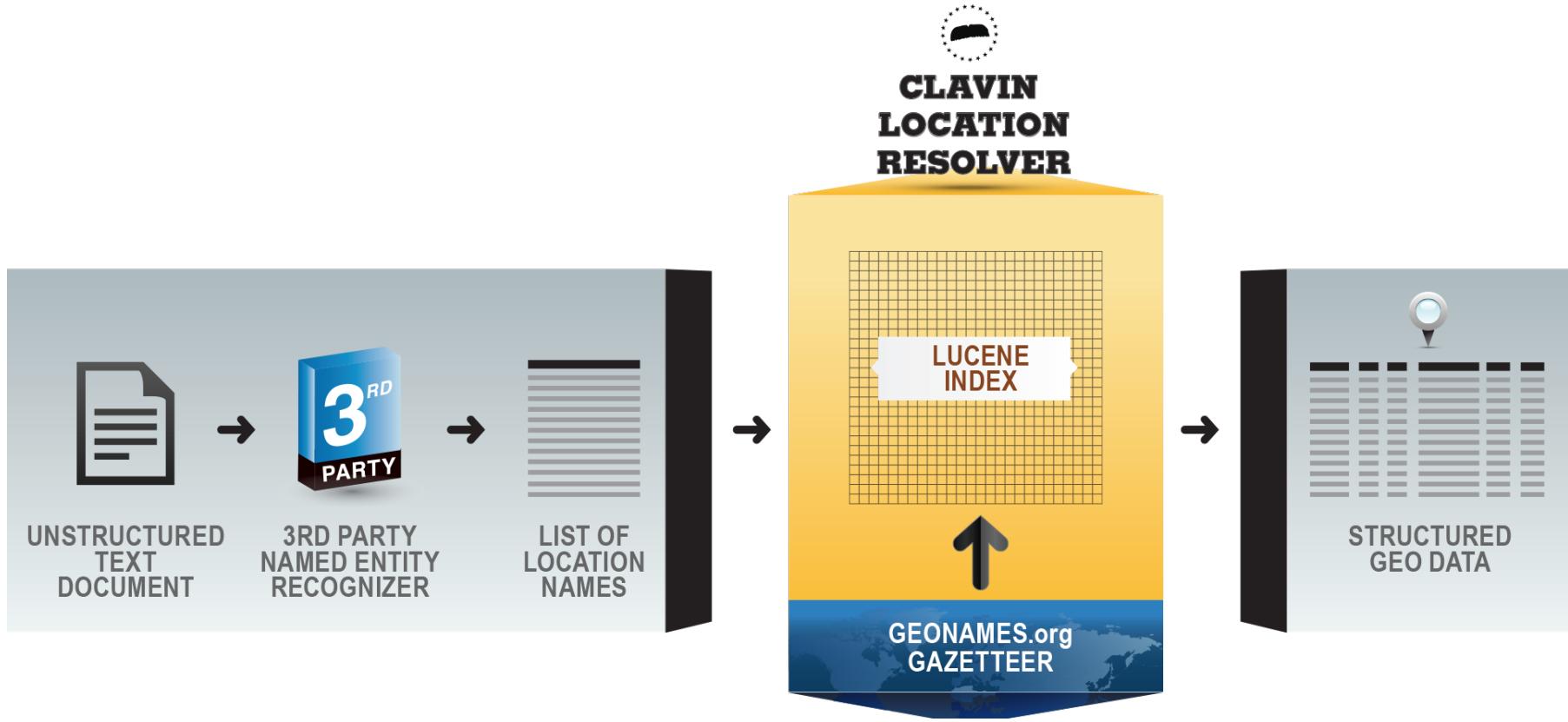
Dealing with Ambiguity

- ◆ **Intelligent Context-based Heuristics**
 - ◆ First: rank by population
 - ◆ Next: look for other locations mentioned in the same document
 - ◆ “Springfield” + “**Chicago**” = Illinois
 - ◆ “Springfield” + “**Boston**” = Massachusetts
 - ◆ Soon: calculate distance based on lat/lons
- ◆ **Resolve alternate names to same geospatial entity**
 - ◆ “Ivory Coast” = “Côte d’Ivoire”
- ◆ **Use fuzzy matching to capture misspelled place names**
 - ◆ Including both phonetic spelling & typographical errors

CLAVIN: an open source geoparser



System Architecture



Live Demonstration

The screenshot shows a web browser window titled "CLAVIN Web Application" with the URL "localhost:8080/clavin-web/". The main content area displays a yellow background with a "Berico Technologies" logo and address, a circular "CLAVIN" stamp, and a wooden pen. On the left, there's a map of Southeast Asia with several location markers. A table lists locations extracted from text, and a "Back to Textbox" button is at the bottom.

Berico Technologies
11130 Sunrise Valley Dr.
Reston, VA, 20191
www.bericotechnologies.com

CLAVIN

BERICO TECHNOLOGIES

Locations Extracted and Resolved From Text

Lat, Lon	Country Code	#
7.2575, 124.20361	PH	6
13, 122	PH	4
10, 118.75	PH	2
6.81304, 125.70848	PH	1
13, 122	PH	1
7.16667, 126.33333	PH	1
7.6, 126.4	PH	1
7.6, 126.68333	PH	1
7.91601, 126.27843	PH	1
13, 122	PH	1
7.4525, 126.58417	PH	1

Map data ©2013 Google, MapIT, Tele Atlas

Only the top 20 locations are shown.

Back to Textbox

Live Demonstration

The screenshot shows a web browser window with the title "CLAVIN Web Application" and the URL "localhost:8080/clavin-web/". The main content area displays a yellow background with a map of the Philippines and surrounding regions. A large, semi-transparent dark gray speech bubble contains the text "What can I do with this data?". Below the map, there is a table of location data:

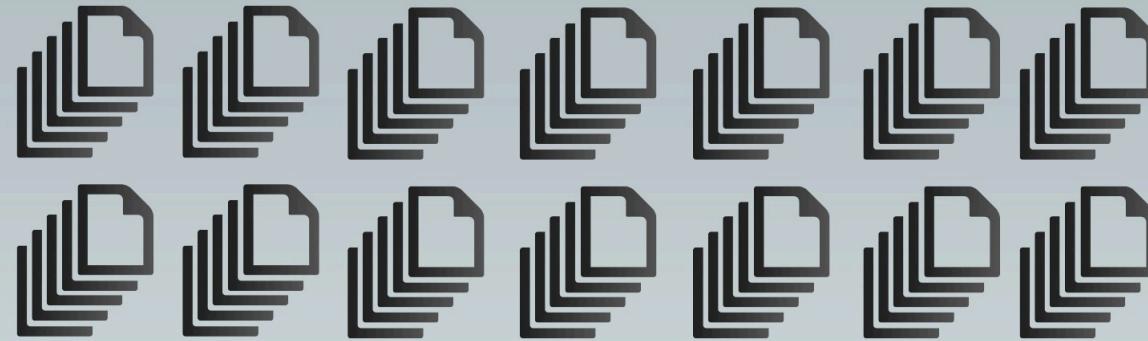
Location	Coordinates	Count
Philippines	6.81304, 13, 122	2
Southeast	7.16667, 13.33333	1
Philippines	7.6, 126.4	1
Philippines	7.6, 126.68333	1
Philippines	7.91601, 126.27843	1
Philippines	13, 122	1
Philippines	7.4525, 126.58417	1

At the bottom left of the slide is a blue button labeled "Back to Textbox". On the right side, there is a wooden pen and a yellow envelope with a postmark that reads "CLAVIN BERICO TECHNOLOGIES".

Map Visualizations



Hierarchical Geospatial Search



Virginia



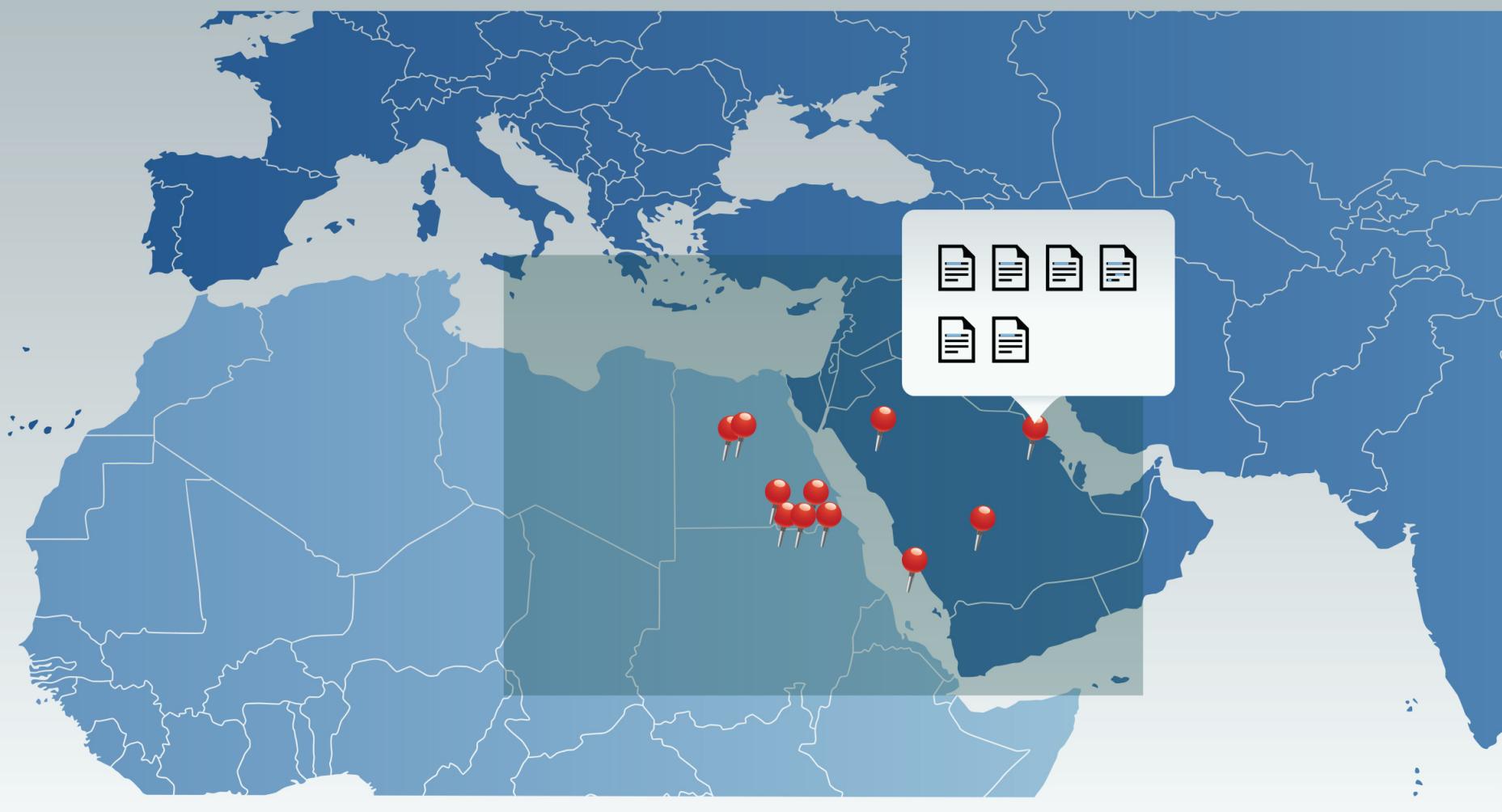
Search Results:

Reston

Arlington



Geospatial Bounding Box Search



Geospatial Analytics on Unstructured Text

Roll up: COUNTRY

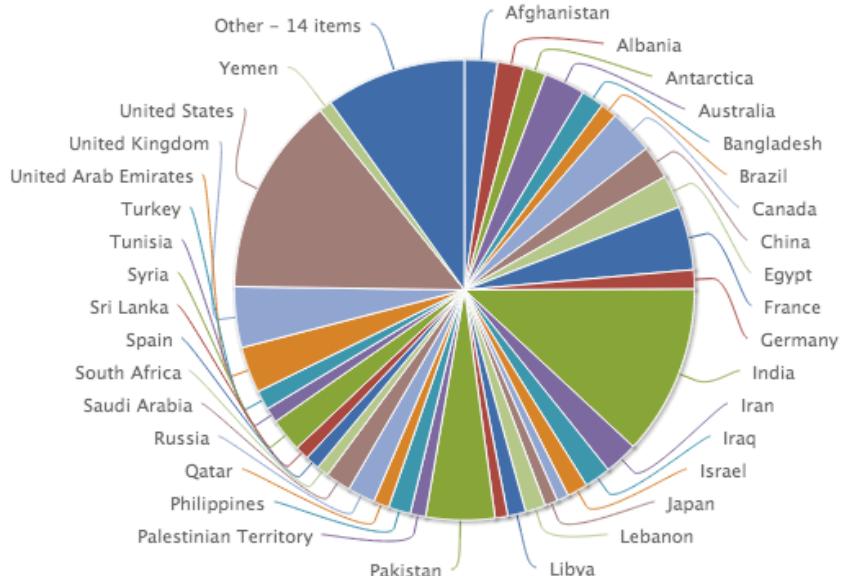
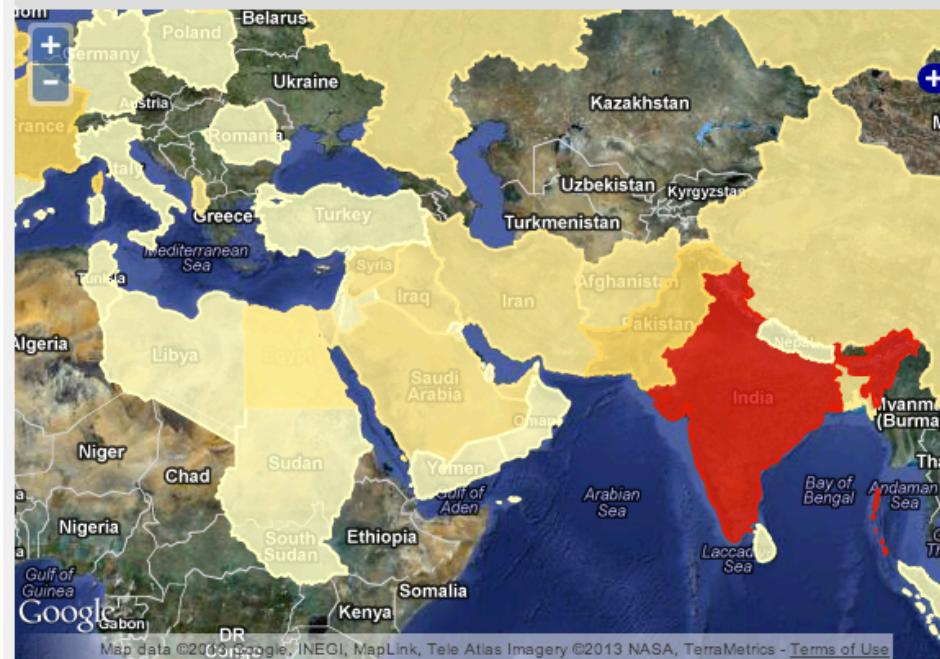
of Values
50

Average
70997

Maximum
496533
United States

Minimum
20761
Sudan

Midpoint
46424
Libya and Turkey



Performance Metrics & Features



CLAVIN

*Cartographic
Location
And
Vicinity
INdexer*

- ◆ **Accurate:** 0.75 F-measure
- ◆ **Fast:** 100 locations per sec per cpu
- ◆ **Scalable:** processes 1M documents in 1 hour on a 9-node Hadoop cluster
- ◆ **Smart:** natural language processing, intelligent heuristics, & fuzzy matching
- ◆ **Easy to use:** simple Java-based API
- ◆ **Open source:** Apache License



clavin.bericotechnologies.com

Charlie Greenbacker
@greenbacker