Geographical Information Retrieval from Text Searching

CS410 - Technology Review
Fall 2018

Team Members

- Zutao Yang - Coordinator (zutaoy2@illinois.edu)
- Nathan Nard (nnard2@illinois.edu)
- Shaun Phillips (shaunap2@illinois.edu)

1. Team Contribution

All of the authors designed the topic together and made significant contributions to this technical review. Zutao Yang led the introduction, section 3 on web mapping sites, and the summary. Nathan Nard led section 4 on ways of extracting text based metadata from images. Finally, Shaun Phillips led section 5 on Python libraries that support both geocoding and geoparsing.

2. Introduction

Geographical information (geo-referenced information or locational information) is recorded in a wide variety of media and document types in the World Wide Web (WWW). There are innumerable articles, books, reports, images maps, databases, and web pages containing text that including information of location. Because everything does takes place in a geographical context, it is not surprising that many information the WWW is location-related, and many queries submitted to search engines have a geographical focus. Due to these facts, attention has been paid in recent years to retrieve geographically specific information from the relatively unstructured but immense documents that compose the World Wide Web [1,e.g., 2,3]. This new technology is called geographical information retrieval (GIR), which can be seen as a specialized branch of traditional information retrieval, with a focus on addition of spatial and geographical indexing and retrieval. In this limited review, we review Web mapping sites as important resources of GIR (**Section 3**), and applications and technologies of GIR from Image Metadata (**Section 4**), and a few important python tools/libraries for GIR (**Section 5**).

3. Web mapping sites

The Geographic World Wide Web or the “GeoWeb” has introduced dramatic change in the number of users and more importantly in the nature of applications of geographic information query, browsing, search, collection, and delivery in the last 15 years, mainly due to the populous of various Web mapping site [4]. As an indicator of this change, the Web mapping sites such as Google Map, Google Earth, Open Street Map (OSM), Bing Maps Baidu Maps (China), MultiMap (United Kingdom) all has attracted millions of visitors every year.

These mapping sites provided geographical database in the form of geo-tagged points, lines, polygons, photos, images, and videos [5]. Geo-tagging combining with a

coordinated system allows these sites to provide geo-referenced query and search. Geotagging-enabled information therefore can be used to find location-based news, websites, or other resources. For example, a user's search with keyword "Gas Station" will return all gas stations within certain geographical area, which is either pre-defined as a default area or could be re-defined by the user. Geotagging is the process of adding geographical information to various media in the form of metadata. The data usually consists of coordinates like latitude and longitude, but may also include place names and other features of the place, like the functionalities (e.g., it is a school). Geo-tagged texts help people get a lot of specific information both about "where is what" and "what is where", through methods and techniques of common text analysis and mining, for example, text parsing, filtering, classification, and clustering, document ranking and evaluation. The ambiguity place names and the vague geographic terminology are special issues that has to be faced using these mapping sites to do GIR [6]. These is because place names in geo-tags can be used to refer to places on Earth, but they may also use for the names of organizations and people or used to describe other facts, and many places may share the same names. For example, a photo tagged with Washington may refer to a photo taken in Washington D.C or may refer to a photo taken for the president Washington D.C, and the Washington may refer to the state of Washington or Washington D.C. To solve such issues, text analysis with geo-parsing and Named Entity Recognition that is a standard part of linguistic analysis in Natural Language Processing (NLP) can be used to identify the presence of genuine place names [3,6], and natural language qualitative spatial relations can be used to understand vague place names [7].

Worldwide, the Google Map/Earth is the most popular mapping site provided by commercial corporation while OSM is probably the most popular grass-rooted mapping site, both providing desktop and mobile web mapping service application. Google Map/Earth offers satellite imagery, street maps, and street view perspectives, layers of different spatial features with geo-tagging, as well as many geo-tagged photos. OSM does not provide many satellite imageries, but on the other hand, offering a huge amount of user-contributed geographical information in the from of tagged vector shapes. While these two popular sites had different motivations in the beginning, they have both ended up important engines for direct geo-referenced search. Users can use key word to search geo-referenced contents. The site will return a ranked list of relevant items, and users can navigate to the selected destination. However, more important GIR application of both sites is assisted by using their Application Programming Interfaces (API) and web crawling tools [8]. Google launched the Google Maps API in June 2005 to allow users to integrate Google Maps into their websites and access and retrieve geo-referenced information through services for text-based search, retrieving static map images, performing encoding, and obtaining geo-tagged photos, and so on. As OSM is released with an open-content license, OSM has an Editing API for fetching and saving raw geodata from/to the OpenStreetMap database, which allow users to not only query and retrieve, but also allow users to edit and contribute. Alternatively, OSM also offers the Overpass API which provides read-only API access. The open-content license and the huge amount of volunteered geographical contents also stimulated the develop of many tools for retrieving geo-referenced data through geo-tags and for analyzing geo-tags. These include installable applications such as Osmosis

(https://wiki.openstreetmap.org/wiki/Osmosis), and web-based tools such as overpass turbo (https://overpass-turbo.eu/), the LinkedGeoData (https://linkedgeodata.org), and the Taginfo (https://taginfo.openstreetmap.org/),the Geofabrik (http://download.geofabrik.de/), the BBBike (https://extract.bbbike.org/), and the Hot export (https://export.hotosm.org/en/v3/), and plugins to mature GIS software such as QuickOSM, QSM Downloader, and QGIS Vector Menu for QGIS, and Qsm2pgsql and Imposm for PostGIS. On the contrary, tools for extracting or scraping geo-referenced content from Google Map/Earth is very few, probably due to its commercial restriction. We only found the Google Maps Scraper (http://www.leadsjack.com/google-maps-scraper/) and G-Business Extractor (http://www.estrattoredati.com/local-business-extractor.html) which provide services to extract customized geo-information from Google Map, but both are not free for users.

The retrieved geo-referenced information from the mapping site Google Map/Earth and OSM has been used for a wide range of applications. In this short review, a complete cover of all the potential applications is out of the scope, but instead we give a few examples. Aside from many business usages, it has been used for collecting of Land Use Land Cover Samples for validating Land Use Land Cover classifications using remote sensing images [9], for indoor mapping for the goal Smart City design[10], for delivering location-based environmental information [11], and for developing web-based tourist information [12], generating web-based 3D geospatial models[13,14], for area-based survey collection [15], for disaster early warning and evaluation [16], for cartographic visualization at street level [17], and for studying street network evolution [e.g., 18], and for topic mapping [e.g., 15] and so on.

4. Image Metadata

Image file formats are capable of storing a rich array of information about the image itself in addition to the information that actually encodes the image's pixels. The extra information they can store are often referred to as "metadata", and there are many different types and formats that an image can contain. Some common examples include IPTC, XMP, Exif, and many others. All of them are defined and disseminated by various organizations and many share common information and formats. Some information they can include are timestamps for when an image was taken, make and model of the device that made the image, name of the user who took the image, many technical and esoteric details about how the image was made, and, especially of interest, geographical information. IPTC is primarily used to caption and categorize photos[19], originally with photojournalists and news organizations in mind. It's gone through a few different revisions over the years and now conforms to the XMP format. XMP is more general and now permits users to create their own custom fields in an image's metadata.Exif primarily contains information about how and where the image was taken[20]. All of these pieces of information provide useful insight into the creation of the image, enabling users to further characterize the image beyond what is visible within the image itself[21].

The geographical information itself has proven to be very useful in the recent decade. Social media websites like Facebook and Instagram have been using image locational

information from their users to help suggest potential friends or direct locally relevant content to their users. Image collecting and sharing websites like Flickr use geolocation to help geographically organize uploader's images so they may be browsed by location and user provided topic tags[22]. More niche uses have been found by those fighting human trafficking, using image metadata to help identify potential human traffickers on the internet[23,24]. Indeed, there's a wealth of utility contained within image metadata and so with the increased volume of images on the internet have come needs for tools for extraction and manipulation of image metadata.

Many social media sites, image sharing sites, and others strip or modify user's images of their metadata unfortunately, so a small percentage of images on the internet contain their metadata. The reasons for stripping or modifying user's images' metadata vary, common reasons include privacy and performance optimization [25]. When collecting large amounts of images, having fast and reliable tools to sift through large collections of images to find metadata is crucial. Fortunately many tools have been developed to help find and extract metadata from images such as software libraries: libexif for C [26], Exiv2 for C++ [26], Pillow for Python [27], and ExifTool for Perl [28]; as well as image site APIs such as those provided by Flickr. Of the software libraries, ExifTool by Phil Harvey stands out as one of the most robust and feature rich toolkits for reading, writing, and creating image metadata[29]. Not only are its tools available for Perl scripting, the tools can also be accessed via a terminal or command line for on the fly metadata reading and manipulation. ExifTool's features include: exhaustive compatibility with image file and metadata formats, supports multiple languages, geotag images based on GPS track logs, generate GPS track logs from geotagged images, account for clock drift between camera and GPS devices to synchronize timestamps, and can recognize a large number of different tag labels among many other features. While the other software packages provide similar functionalities, ExifTool appears to be the "State of the Art" in image metadata extraction and manipulation. Many web applications that view image metadata, such as Jeffrey's Exif Viewer, use ExifTool as the backend and many wrapper libraries have been authored for other programming languages,[30] such as pyexiftool for python and exiftoolr for ruby. The other software libraries mentioned above provide similar functionalities but not as robust as ExifTool [28,31]. Using any of these libraries, one can extract geographical information from images and use that information in sites like OSM to obtain nearby mailing addresses for a location as well as other relevant information about the area.

One excellent source for images containing metadata is Flickr, which organizes users' image uploads by subject as well as geolocation (if provided). Flickr also provides an API for searching for images based on geolocation, image id, and other tag information. Conveniently, many different wrapper toolkits have been developed for the API within many different programming languages, such as Beej's Python Flickr API kit. Search results are returned in XML format which then the user must parse to find and understand the relevant information returned by the API[32]. Using Flickr's API, it is possible to retrieve information on all images on their site that contain a tag of interest, like "dam" for example, that also contain geographical information. With this geographical information one can pinpoint potential locations of dams. However, such a search returns thousands of image results that are impossible to sift through by

hand to confirm whether or not they represent actual physical dams. So then further analysis will be needed to help identify actual dams, i.e., one could use NLP on the images' other text fields (on Flickr) to gain better insight into the nature of the image. Or one could also use the geolocation information and see if other web services, like OSM, corroborate whether or not the images correspond to actual dams.

As presented, it can be seen that there are many potentials uses for image metadata. Many organizations—such as Facebook, Instagram, and Flickr—use image metadata, particularly geographical information, to better enhance their customer's user experience. At the same time however are issues of privacy, causing nearly widespread removal of image metadata making it difficult to find any images with useful information. However, with the tools available today, for extracting and manipulating image metadata as well as sharing and searching images, anyone can find any and all valuable information from large collections of images. From locating important landmarks to mapping individual's travel patterns, the utilities appear to be numerous and broad.

5. Important Python Tools and Libraries for GIR

A large portion of web search traffic is GIR related. In 2003 alone, the volume of web searches related to GIR was in the range of 13-15% [33]. In order to support converting user queries into a form usable for GIR involves two major steps. The first is ***geoparsing***, a form of Named Entity Recognition (NER) specifically concerned with parsing descriptions of places from text and converting them into specific place names, also known as toponyms. The next is ***geocoding***, which is concerned with converting a place name into an actual physical location as specified by a set of geographic coordinates, usually using a gazetteer [34].

Python provides a rich set of libraries supporting the full geospatial work flow, which includes geoparsing, geocoding, and rich data visualization. This section discusses libraries for geoparsing and geocoding. These libraries directly support performing GIR through either direct user queries or as part of an automated workflow to mine free form web data and populate a datastore of geographic entities of interest. Several references are available that discuss how to perform visualization of geographic data using Python and a few examples are included here for completeness but will not be discussed [35,36].

The following Python geoparsing libraries apply logic and matching rules in order to correctly parse the names of places, locations of interest, streets, addresses, cities, counties, states, and countries. The input is generally a text string or document link and the output is a collection of toponyms.

The first library for geoparsing is Geography [37]. This library is a wrapper around NLTK, a platform for natural language processing [38]. Geography is only concerned with parsing place names from text and providing this information in an easy to consume format. For example, the response specifically groups entities based on countries, regions, cities, and other. Other is used to refer to those items that are places, but which do not explicitly map to one of these categories. Additionally,

Geography can group place names based on country and provide counts of each group. Usage is straightforward as a text string or URL to a web page is specified and the library automatically parses entities. Accuracy is directly related to the accuracy of the wrapped NLTK library for performing entity recognition. Potential downsides to the usage of this library include installation of multiple dependencies, infrequent code updates, multiple project versions, and no clear owner across multiple GitHub repositories.

Another library which provides similar functionality to Geography is Geotext [38]. The main differences of Geotext are that it has no external dependencies and it is relatively fast. Speed comes from not using NLTK but instead using the Python built in regular expression processing library in conjunction with a static cache of geographic data provided by www.geonames.org. Therefore, Geotext is not as feature rich as Geography but compensates with a reduced installation footprint and faster run time operation. An interesting feature of Geotext is the ability to specify a specific country code when passing in text for processing which automatically filters the response to only include matching cities for that country. Disadvantages of using this library are that it has a small committer base and infrequent commits.

The last geoparsing library discussed is spaCy [38]. spaCy is a full natural language processing library that provides strong support for NER extraction. It provides similar functionality to NLTK but with a different design philosophy. NLTK is designed to support teaching and research. spaCy is designed for use in building real world applications that are as performant as possible and where key decisions have already been made for the developer. It provides an optimal way of doing things instead of multiple ways of doing things. The feature set is rich and the provided documentation and code examples are robust. Additionally, Spacy comes with a visualizer tool for dynamically examining the extracted entity output. NER extraction provides built in support for FAC, GPE, and LOC entity types. FAC represents buildings, airports, bridges, and other facilities identifiable by name. GPE stands for geopolitical entity and represents cities, states, and countries. Finally, LOC stands for locations that are not GPE based, for example, bodies of water, mountains, and other natural features. The FAC and LOC extraction features would be especially useful for extracting dams and other structures in the context of the bodies of water on which they are located. The Spacy library is actively developed with a high count of both contributors and forks on GitHub.

The next section discusses Python geocoding libraries which convert named locations into geographic coordinates. These libraries typically provide a standard interface to multiple online geolocation service providers and act as convenience wrappers. The key advantage of these geocoding libraries is their ability to extract away API specific details for each service and provide a consistent end user experience during application development. However, since some providers may support a richer feature set beyond simple coordinate look up, it is imperative to fully investigate the documentation of each library to understand what additional features for each provider are included in the wrapper implementation.

Geopy supports a total of 22 geocoder providers [38]. The project is actively

developed with a high volume of forks and commits on GitHub. Geopy supports both standard look ups as well as reverse lookups, where specific latitude and longitude coordinates are provided instead of a location name or address and the service attempts to provide the best address that maps to the coordinates.

Geocoder provides similar functionality to Geopy, and the size and support level of the projects appear similar based on GitHub statistics [38]. One additional feature Geocoder supports is the ability to attempt coordinate lookup based on IP address if supported by the specific geocoder provider. This functionality is useful for attempting to identify the physical location of a web resource when no other information is known. Also, at least for some of the providers, the ability to perform batch requests is supported. This mitigates API request load to remote servers and can enable performance speed ups for applications that make higher volumes of requests. A total of 28 providers are supported.

6. Summary

In summary, in this brief review, we have included most popular mapping sites that integrated important text-based geographical information that is integrated with both raster features (images and photos) and vector features (points, lines, and polygons). We also included important concepts, platforms, and tools to retrieve geographical information based on metadata of photos and images. Lastly, we reviewed popular python library and tools that could be more widely customized to crawl and retrieve geographical information from the huge amount of text in the Internet.

References

1. Larson, R.R.; Frontiera, P. Geographic Information Retrieval and Spatial Browsing. In Proceedings of the 32nd Clinic on Library Applications of Data Processing; 1996.
2. Egenhofer, M.J. Toward the semantic geospatial web. In Proceedings of the Proceedings of the tenth ACM international symposium on Advances in geographic information systems - GIS '02; 2002.
3. Purves, R.S.; Clough, P.; Jones, C.B.; Arampatzis, A.; Bucher, B.; Finch, D.; Fu, G.; Joho, H.; Syed, A.K.; Vaid, S.; et al. The design and implementation of SPIRIT: A spatially aware search engine for information retrieval on the Internet. *Int. J. Geogr. Inf. Sci.* **2007**, doi:10.1080/13658810601169840.
4. Haklay, M.; Singleton, A.; Parker, C. Web mapping 2.0: The neogeography of the GeoWeb. *Geogr. Compass* **2008**, doi:10.1111/j.1749-8198.2008.00167.x.
5. Zheng, Y.T.; Zha, Z.J.; Chua, T.S. Research and applications on georeferenced multimedia: A survey. *Multimed. Tools Appl.* **2011**, doi:10.1007/s11042-010-0630-z.
6. Jones, C.B.; Purves, R.S. Geographical information retrieval. *Int. J. Geogr. Inf. Sci.* **2007**, doi:10.1080/13658810701626343.
7. Cai, G.; Wang, H.; MacEachren, A. Communicating Vague Spatial Concepts in Human-GIS Interactions: A Collaborative Dialogue Approach. *Spat. Inf. Theory* **2003**, doi:10.1002/pros.20696.
8. Bošnjak, M.; Oliveira, E.; Martins, J.; Mendes-Rodrigues, E.; Sarmento, L. TwitterEcho - A Distributed Focused Crawler to Support Open Research with Twitter Data. *Proc. WWW 2012, 21st Int. Conf. Companion World Wide Web* **2012**, doi:10.1145/2187980.2188266.
9. Hou, D.; Chen, J.; Wu, H.; Li, S.; Chen, F.; Zhang, W. Active collection of land cover sample data from geo-tagged web texts. *Remote Sens.* **2015**, doi:10.3390/rs70505805.
10. Amat, G.; Fernandez, J.; Ramos, A. Using Open Street Maps data and tools for indoor mapping in a Smart City scenario. *Agile* **2014**.
11. Ciepluch, B.; Mooney, P.; Jacob, R.; Winstanley, A.C. Using OpenStreetMap to deliver location-based environmental information in Ireland. *SIGSPATIAL Spec.* **2009**, *1*, 17, doi:10.1145/1645424.1645428.
12. Pan, B.; Crotts, J.C.; Muller, B. Developing Web-Based Tourist Information Tools Using Google Map. In *Information and Communication Technologies in Tourism 2007*; Springer Vienna: Vienna, 2007; pp. 503–512.
13. Over, M.; Schilling, A.; Neubauer, S.; Zipf, A. Generating web-based 3D City Models from OpenStreetMap: The current situation in Germany. *Comput. Environ. Urban Syst.* **2010**, *34*, 496–507, doi:10.1016/J.COMPENVURBSYS.2010.05.001.
14. Akanbi, A.K.; Agunbiade, O.Y. Integration of a city GIS data with Google Map API and Google Earth API for a web based 3D Geospatial Application. **2013**.
15. Bearman, N.; Appleton, K. Using Google Maps to collect spatial responses in a survey environment. *Area* **2012**, *44*, 160–169, doi:10.1111/j.1475-4762.2012.01081.x.
16. Rahman, K.M.; Alam, T.; Chowdhury, M. Location based early disaster warning and evacuation system on mobile phones using OpenStreetMap. In

Proceedings of the 2012 IEEE Conference on Open Systems, ICOS 2012; 2012.
17. Gibin, M.; Singleton, A.; Milton, R.; Mateos, P.; Longley, P. An Exploratory Cartographic Visualisation of London through the Google Maps API. *Appl. Spat. Anal. Policy* **2008**, doi:10.1007/s12061-008-9005-5.
18. Neis, P.; Zielstra, D.; Zipf, A. The Street Network Evolution of Crowdsourced Maps: OpenStreetMap in Germany 2007–2011. *Futur. Internet* **2011**, doi:10.3390/fi4010001.
19. Types Of Metadata Available online: https://www.photometadata.org/META-101-metadata-types.
20. PTC Photo Metadata Standard Available online: https://iptc.org/standards/photo-metadata/iptc-standard/.
21. EXIF Data Available online: http://www.geofflawrence.com/photography_tutorial_exif_data.html.
22. Exposing the Invisible.
23. Spyrou, E.; Mylonas, P. Analyzing Flickr metadata to extract location-based information and semantically organize its photo content. *Neurocomputing* **2016**, doi:10.1016/j.neucom.2014.12.104.
24. Mattmann, C.; Yang, G.H.; Manjunatha, H.; Gowda, T.; Zhou, A.J.; Luo, J.; McGibbney, L.J. Multimedia Metadata-based Forensics in Human Trafficking Web Data 2016.
25. State of image metadata in 2018 Available online: https://imatag.com/en/blog/2018/05/11/state-of-image-metadata-in-2018/.
26. The libexif C EXIF library Available online: https://libexif.github.io/.
27. Pillow Available online: https://pillow.readthedocs.io/en/5.3.x/.
28. ExifTool Available online: https://github.com/mceachen/exiftoolr.
29. Flickr Available online: https://www.flickr.com/services/api/.
30. Jeffrey Friedl's Blog Available online: http://regex.info/blog/other-writings/online-exif-image-data-viewer.
31. PyExifTool – A Python wrapper for Phil Harvey's ExifTool Available online: http://smarnach.github.io/pyexiftool/.
32. Flickrapi Available online: https://stuvel.eu/flickrapi.
33. Rüger, S. Multimedia Information Retrieval. *Synth. Lect. Inf. Concepts, Retrieval, Serv.* **2009**, doi:10.2200/S00244ED1V01Y200912ICR010.
34. Moura, T.H.V.M.; Davis, C.A.; Fonseca, F.T. Reference data enhancement for geographic information retrieval using linked data. *Trans. GIS* **2017**, doi:10.1111/tgis.12238.
35. Garrard, C. (Christine M.. *Geoprocessing with Python*; ISBN 9781617292149.
36. Westra, E. *Python geospatial development : build a complete and sophisticated mapping application from scratch using Python tools for GIS development*; 2010; ISBN 9781849511544r1849511543.
37. Geograpy Available online: https://github.com/ushahidi/geograpy.
38. Natural Language Toolkit Available online: http://www.nltk.org.