A Survey on Geocoding: Algorithms and Datasets for Toponym Resolution

Anonymous ACL submission

Abstract

Geocoding, the task of converting unstructured text to structured spatial data, has recently seen progress thanks to a variety of new datasets, evaluation metrics, and machine-learning algorithms. We provide a comprehensive survey to review, organize and analyze recent work on geocoding (also known as toponym resolution) where the text is matched to geospatial coordinates and/or ontologies. We summarize the findings of this research and suggest some promising directions for future work.

1 Introduction

Geocoding, also called toponym resolution or toponym disambiguation, is the subtask of geoparsing that disambiguates place names in text. The goal of geocoding is, given a textual mention of a location, to choose the corresponding geospatial coordinates, geospatial polygon, or entry in a geospatial database. Geocoders must handle place names (known as *toponyms*) that refer to more than one geographical location (e.g., *Paris* can refer to a town in the state of *Texas* in the *United States*, or the capital city of *France*), and geographical locations that may be referred to by more than one name (e.g., *Leeuwarden* and *Ljouwert* are two names for the same city in the Netherlands), as shown in fig. 1. Geocoding plays a critical role in tasks such as tracking the evolution and emergence of infectious diseases (Hay et al., 2013), analyzing and searching documents by geography (Bhargava et al., 2017), geospatial analysis of historical events (Tateosian et al., 2017), and disaster response mechanisms (Ashktorab et al., 2014; de Bruijn et al., 2018).

Prior surveys (Leidner, 2007; Gritta et al., 2017) predate modern neural network approaches to geocoding. The number of geocoding datasets has also doubled since 2017, and new geocoding algorithms have been introduced with each new dataset. The field would thus benefit from a comprehensive survey and critical evaluation of the currently available datasets, evaluation metrics, and geocoding algorithms. Our contributions are:

- the first survey to review deep learning approaches to geocoding
- comprehensive coverage of geocoding systems, which increased by 50% in the last 4 years
- comprehensive coverage of annotated geocoding datasets, which increased by 100% in the last 4 years

2 Background

An early work on geocoding, Amitay et al. (2004), identifies two important types of ambiguity: A place name may also have a non-geographic meaning, such as *Turkey* the country vs. *turkey* the animal, and two places may have the same name, such as the *San Jose* in California and the *San Jose* in Costa Rica. The former is a challenge during geotagging, i.e., finding the place names, while the latter is a challenge during geocoding, i.e., matching the place names to geospatial coordinates, geospatial polygons, or entries in a geospatial database.

To the best of our knowledge, the first formal survey of geocoding is the PhD thesis of Leidner (2007). The thesis found that most geocoding methods were based on combining natural language processing techniques, such as lexical string matching or word sense matching, with geographic heuristics, such as spatial-distance minimum and population maximum. Most geocoders studied in this thesis were rule-based.

Gritta et al. (2017) reviewed both geotagging and geocoding, while proposing a new dataset, **WikToR**. The survey portion of this article compared a variety of datasets for geoparsing, explored heuristics of rule-based and feature-based machine learning-based geocoders, summarized evaluation metrics, and classified common errors cases from

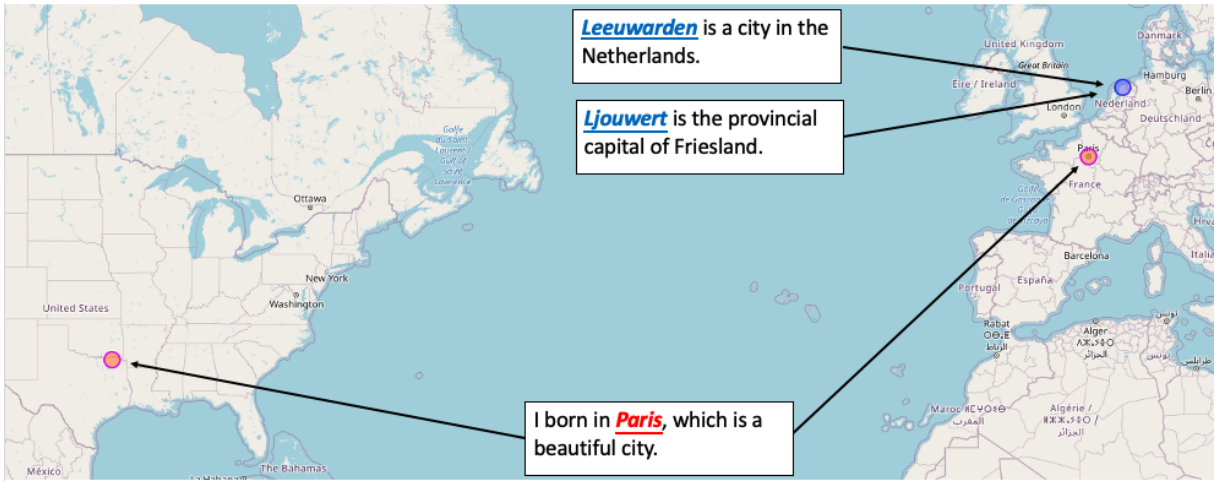

Figure 1: An illustrative example of geocoding challenges. One toponym (*Paris*) can refer to more than one geographical location (a town in the state of *Texas* in the *United States* or the capital city of *France* in *Europe*), and a geographical location may be referred to by more than one toponym (*Leeuwarden* and *Ljouwert* are two names for the same city in the Netherlands).

several geocoders (misspellings, case sensitivity, processing fictional and historical text presents, etc.). Gritta et al. (2017) concluded that future geoparsers would need to utilize semantics and context, not just syntax and word forms as the geocoders of the time.

Geocoding research since these previous surveys has changed in several important ways. It has indeed incorporated more semantics, especially through new deep learning methods. It has pivoted from generating database entries to directly generating geospatial polygons. And some of the most recent work goes beyond explicit place names to inferring polygons for place descriptions like *between the towns of Adrano and S. Maria di Licodia* (Laparra and Bethard, 2020).

3 Methodology

We searched for literature using the Google Scholar and Semantic Scholar search engines. We retrieved papers matching any of the keyword queries: *geocoding*, *geoparsing*, *geolocation*, *toponym resolution*, *toponym disambiguation*, or *spatial information extraxtion*. We manually excluded papers that matched these keywords but were not about geocoding. For example, we excluded papers that tried to match a full document or microblog post to a single location, rather than trying to match each individual place name to a location. We also filtered most papers published before 2010, as they have been covered thoroughly by prior surveys, though we still included influential older papers that were highly cited by the papers we collected via the process above. In total, we reviewed more than 60 papers and included more than 30 of them in this survey.

4 Geocoding Datasets

Many geocoding corpora have been proposed, drawn from different domains, linking to different geographic databases, with different forms of geocoding labels, and with varying sizes in terms of both articles/messages and toponyms. Table 1 summarizes these datasets, and the following sections walk through some of the dimensions over which the datasets vary.

4.1 Domains

The news domain is the most common target for geocoding corpora, covering sources like broadcast conversation, broadcast news, and news magazines. Examples include the ACE 2005 English SpatialML Annotations (ACS, Mani et al., 2010)[1], the Local Global Lexicon (LGL, Lieberman et al., 2010), CLUST (Lieberman and Samet, 2011), TR-NEWS (Kamalloo and Rafiei, 2018), GeoVirus (Gritta et al., 2018), and GeoWebNews (Gritta et al., 2019). Though all these datasets include news text, they vary in what toponyms are included. For example, LGL is based on local and small U.S. news sources with most toponyms smaller than a U.S.

[1] https://catalog.ldc.upenn.edu/LDC2008T03 https://catalog.ldc.upenn.edu/LDC2011T02

Corpus	Domain	Geographic Database	Label Type	Articles / Messages	Toponyms
ACS, Mani et al. (2010)	News	GeoNames	Point	428	4783
LGL, Lieberman et al. (2010)	News	GeoNames	Point & GeoNamesID	588	4783
CLUST, Lieberman and Samet (2011)	News	GeoNames	Point & GeoNamesID	1082	11564
Zhang and Gelernter (2014)	Twitter	GeoNames	Point & GeoNamesID	956	1393
WOTR, DeLozier et al. (2016)	Historical	OpenStreetMap	Point & Polygon	9653	10380
WikTOR, Gritta et al. (2017)	Wikipedia	GeoNames	Point	5000	25000
TR-NEWS, Kamalloo and Rafiei (2018)	News	GeoNames	Point & GeoNamesID	118	1274
GeoCorpora, Wallgrün et al. (2018)	Twitter	GeoNames	Point & GeoNamesID	211	2966
GeoVirus, Gritta et al. (2018)	News	GeoNames	Point	229	2167
GeoWebNews, Gritta et al. (2019)	News	GeoNames	Point & GeoNamesID	200	5121
SemEval2019, Weissenbacher et al. (2019)	Scientific	GeoNames	Point & GeoNamesID	150	8360
GeoCoDe, Laparra and Bethard (2020)	Wikipedia	OpenStreetMap	Polygon	360187	360187

Table 1: Summary of geocoding datasets covered by this survey, sorted by year of creation.

state, while GeoVirus focuses on news about global disease outbreaks and epidemics with larger, often country-level, toponyms.

Web text is also a common target for geocoding corpora. Wikipedia Toponym Retrieval (WikToR; Gritta et al., 2017) and GeoCoDe (Laparra and Bethard, 2020) are both based on Wikipedia pages. ACS, mentioned above, also includes newsgroup and weblog data. And social media, specifically Twitter, is the target for the Zhang and Gelernter (2014) dataset and GeoCorpora (Wallgrün et al., 2018). These corpora vary as widely as the internet text upon which they are based. For example, GeoCoDe and WikToR include the first paragraphs of Wikipedia articles, while Zhang and Gelernter (2014) and GeoCorpora contain Twitter messages with place names that were highly ambiguous and mostly unambiguous, respectively.

Other geocoding domains are less common, but have included areas such as historical documents and scientific journal articles. The Official Records of the War of the Rebellion (WOTR; DeLozier et al., 2016) corpus annotates historical toponyms of the U.S. Civil War. The SemEval-2019 Task 12 dataset (Weissenbacher et al., 2019) is based on scientific journal papers from PubMed Central[2].

4.2 Geographic Databases

All geocoding corpora rely on some database of geographic knowledge, sometimes also called a gazetteer or ontology. Such a database includes canonical names for places along with their geographic attributes such as latitude/longitude or geospatial polygon, and may include other information, such as population or type of place.

Most geocoding corpora have used GeoNames[3] as their geographic database, including ACS, LGL, CLUST, the Zhang and Gelernter (2014) corpus, WikToR, TR-NEWS, GeoCorpora, GeoVirus, GeoWebNews, and the SemEval-2019 Task 12 corpus. GeoNames is a crowdsourced database of geospatial locations, with almost 7 million entries and a variety of information such as feature type (country, city, river, mountain, etc.), population, elevation, and positions within a political geographic hierarchy. The freely available version of GeoNames contains only a (latitude, longitude) point for each location, with the polygons only available with a premium data subscription, so most corpora based on GeoNames do not use geospatial polygons.

Geocoding corpora where recognizing geospatial polygons is important have typically turned to OpenStreetMap[4]. OpenStreetMap is another crowdsourced database of geospatial locations, which contains both (latitude, longitude) points and geospatial polygons for its locations. WOTR and GeoCoDe are based on OpenStreetMap.

4.3 Geospatial Label Types

Three different types of geospatial labels have been considered in geocoding corpora: database entries, (latitude, longitude) points, and polygons. All corpora except WTOR and GeoCoDe assign to each place name the (latitude, longitude) point that represents its geospatial center on the globe. Many of the GeoNames-based corpora (LGL, CLUST, TR-NEWS, GeoCorpora, GeoWebNews, and the SemEval-2019 Task 12 corpus) also assign to each place name its GeoNames database ID. The WTOR corpus assigns to each place name a point or a poly-

[2] https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

[3] https://www.geonames.org/

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

Figure 2: The red-shaded area is the polygon label for *Biancavilla*, which is defined by the set of its boundary coordinates retrieved from OpenStreetMap.

gon, and GeoCoDe assigns to each place name only a polygon. Figure 2 shows an example of a polygon annotation from GeoCoDe.

5 Geocoding Evaluation Metrics

Geocoding systems are evaluated on geocoding corpora using metrics that depend on the corpus's geospatial label type.

5.1 Database entry correctness metrics

When the target label type is a geospatial database entry ID, common evaluation metrics for multi-class classification tasks are applied. These metrics can also be used for corpora with (latitude, longitude) point labels by breaking the globe down into a discrete grid of geospatial tiles, and treating each geospatial tile like a database entry.

Accuracy is the number of place names where the system has predicted the correct database entry, divided by the number of place names. Accuracy is sometimes also called *Precision@1* or *P@1* when there is only one correct answer (as in the case for current geocoding datasets) and when the ranking-based system is turned into a classifier by taking the top-ranked result as its prediction (the current standard for geocoding evaluation).

$$Accuracy = \frac{\left|\hat{U}\right|}{|U|}$$

where U is the set of all human-annotated place names, $\hat{U}$ is the set of place names where the system has predicted correctly or the top-1 ranked result is correct.

5.2 Point distance metrics

When the target label type is a (latitude, longitude) point, common evaluation metrics attempt to measure the distance between the system-predicted point and the human-annotated point.

Mean error distance calculates the mean over all predictions of the distance between each system-predicted and human-annotated point:

$$MeanErrorDist = \frac{\sum_{u \in U} dis(l_s(u), l_h(u))}{|U|}$$

where U is the set of all human-annotated place names, $l_s(u)$ is the system-predicted (latitude, longitude) point for place name u, $l_h(u)$ is the human-annotated (latitude, longitude) point for place name u, and dis is the distance between the two points on the surface of the globe. This metric is sensitive to outliers, and in realistic scenarios the distribution of error distance is not normal. Gritta et al. (2017) found that the bulk of errors are triggered by roughly 20% of the places and the errors from the remaining places are relatively low.

Median Error Distance is defined in a similar way to mean error distance, but takes the median of the error distances rather than the mean. This metric has fewer problems with outliers than mean error distance.

Accuracy@k km/miles measures the fraction of system-predicted (latitude, longitude) points that were less than k km/miles away from the human-annotated (latitude, longitude) points. Formally:

$$Acc@k = \frac{|\{u | u \in U \wedge dis(l_s(u), l_h(u)) <= k\}|}{|U|}$$

where U, l_s, l_h, and dis are defined as above, and k is a hyper-parameter. The value of k may be hard to select, but many researchers have adopted 161 as the value of k, that is using *Accuracy@161 km* (Cheng et al., 2010).

Area Under the Curve (AUC) calculates the area under the curve of the distribution of geocoding error distances. A geocoding system is better if the area under the curve is smaller. Formally:

$$AUC = \ln \frac{ActualErrorDistance}{MaxPossibleErrors}$$

where $ActualErrorDistance$ is the area under the curve, and $MaxPossibleErrors$ is the farthest distance between two places on earth. The

value of *AUC* is between 0 and 1 and the difference between two small errors (such as 10 and 20 km) is more significant than the same difference between two large errors (such as 110 and 120 km). This makes *AUC* more popular than *Accuracy@k km/miles* (Jurgens et al., 2015).

5.3 Polygon-based metrics

When the target label type is a polygon, evaluation metrics attempt to compare the overlap between the system-predicted polygon and the human-annotated polygon.

Polygon-based precision and recall were proposed by Laparra and Bethard (2020) based on the intersection of system-predicted and human-annotated geometries. Formally:

$$Precision = \frac{1}{|S|} \sum_{i \in |S|} \frac{area(S_i \cap H_i)}{area(S_i)}$$

$$Recall = \frac{1}{|H|} \sum_{i \in |H|} \frac{area(S_i \cap H_i)}{area(H_i)}$$

where the S is the system-predicted set of polygons and H is the human-annotated set of polygons.

6 Geocoding Systems

Table 2 summarizes the approaches of geocoders over the last decade. These models have different approaches to the prediction problem, ranging from ranking to classification to regression. They implement their predictive models with technology ranging from hand-constructed rules and heuristics, to feature-based machine-learning models, to deep learning (i.e., neural network) models that learn their own features.

6.1 Prediction Types

Ranking is the most common approach to making geospatial predictions (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Martins et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; MG, Freire et al., 2011; CLAVIN, Berico Technologies, 2012; Lieberman and Samet, 2012; WISTR, Speriosu and Baldridge, 2013; GeoTxt, Karimzadeh et al., 2013; Zhang and Gelernter, 2014; CBH, SHS, Kamalloo and Rafiei, 2018; DM_NLP, Wang et al., 2019). For example, most rule-based systems index their geospatial database with a search system such as Lucene[5], and query that index to produce a ranked list of candidate database entries. This ranked list may be further re-ranked based on other features such as population or proximity. The type of scores using in re-ranking include binary classification score (MG, Freire et al., 2011; Lieberman and Samet, 2012; WISTR, Speriosu and Baldridge, 2013; Zhang and Gelernter, 2014; CBH, SHS, Kamalloo and Rafiei, 2018; DM_NLP, Wang et al., 2019), regression distance (Martins et al., 2010) and heuristics based on information in the geospatial database (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; CLAVIN, Berico Technologies, 2012; GeoTxt, Karimzadeh et al., 2013).

Classification is commonly used in making geospatial predictions when the Earth's surface has been discretized into tiny areas (Topocluster, DeLozier et al., 2015; CamCoder, Gritta et al., 2018; Cardoso et al., 2019; MLG, Kulkarni et al., 2020). For example, *CamCoder* divides the Earth's surface into 7,823 tiles, and then changes the geospatial label of each toponym to the tile containing its coordinate. *CamCoder* then directly predicts one of 7823 classes for each toponym mention.

Regression is sometimes used for geospatial predictions when the label type is a (latitude, longitude) point or a polygon (Cardoso et al., 2019; Laparra and Bethard, 2020). For example, Laparra and Bethard (2020) predict a set of coordinates (i.e., a polygon) by applying operations over reference geometries, where the operations take sets of coordinates as inputs and produce sets of coordinates as outputs. Regression approaches to geocoding are rare because directly predicting coordinates over the entire surface of the Earth is challenging.

6.2 Features and Heuristics

All geocoding systems combine string matching (exact string matching, Levenshtein distance, etc.) with other features and/or heuristics (population, words in nearby context, etc.). Details of such features are described in this section.

String match checks whether the place name matches any names in the geospatial database (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Martins et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; MG, Freire et al., 2011; CLAVIN, Berico Technologies, 2012; GeoTxt, Karimzadeh et al., 2013; Zhang and Gel-

[5] https://lucene.apache.org/

GeoCoder	Implementation	Prediction Type	Database Independent	Polygon based
Edinburgh Parser, Grover et al. (2010)	Rule-based	Ranking	No	No
Tobin et al. (2010)	Rule-based	Ranking	No	No
Martins et al. (2010)	Machine Learning	Ranking	No	No
Lieberman et al. (2010)	Rule-based	Ranking	No	No
Lieberman and Samet (2011)	Rule-based	Ranking	No	No
MG, Freire et al. (2011)	Machine Learning	Ranking	No	No
CLAVIN, Berico Technologies (2012)	Rule-based	Ranking	No	No
Lieberman and Samet (2012)	Machine Learning	Ranking	No	No
GeoTxt, Karimzadeh et al. (2013)	Rule-based	Ranking	No	No
WISTR, Speriosu and Baldridge (2013)	Machine Learning	Ranking	No	No
Zhang and Gelernter (2014)	Machine Learning	Ranking	No	No
Topocluster, DeLozier et al. (2015)	Machine Learning	Classification	Yes	No
CBH, SHS Kamalloo and Rafiei (2018)	Machine Learning	Ranking	No	No
CamCoder, Gritta et al. (2018)	Deep Learning	Classification	No	No
DM_NLP, Wang et al. (2019)	Machine Learning	Ranking	No	No
Cardoso et al. (2019)	Deep Learning	Classification & Regression	Yes	No
MLG, (Kulkarni et al., 2020)	Deep Learning	Classification	Yes	No
Laparra and Bethard (2020)	Rule-based	Regression	Yes	Yes

Table 2: Summary of geocoding systems covered by this survey, sorted by year of creation.

ernter, 2014; CBH, SHS, Kamalloo and Rafiei, 2018; DM_NLP, Wang et al., 2019). String matching can be done exactly, or approximately with edit distances metrics like Levenshtein Distance. For example, GeoTxt calculates the Levenshtein Distance between the place name mentioned in text and each candidate entry from the geospatial database, and selects the candidate with the lowest edit distance.

Population looks at the size of the population associated with candidate database entry, typically preferring more populous entries to less populous ones (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Martins et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; MG, Freire et al., 2011; Lieberman and Samet, 2012; CLAVIN, Berico Technologies, 2012; GeoTxt, Karimzadeh et al., 2013; Zhang and Gelernter, 2014; CBH, SHS, Kamalloo and Rafiei, 2018; CamCoder, Gritta et al., 2018; DM_NLP, Wang et al., 2019). For example, when the Edinburgh Parser geocodes the text *I love Paris. I love France!* it resolves *Paris* to PARIS, FRANCE instead of PARIS, TX, U.S. since the former has a greater population in the geospatial database.

Type of place looks at the geospatial feature type (country, city, river, populated place, facilities, etc.) of a candidate database entry, typically preferring the more geographically prominent ones (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Martins et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; MG, Freire et al., 2011; CLAVIN, Berico Technologies, 2012; Lieberman and Samet, 2012; GeoTxt, Karimzadeh et al., 2013; TRAWL, Speriosu and Baldridge, 2013; Zhang and Gelernter, 2014; CBH, SHS, Kamalloo and Rafiei, 2018; DM_NLP, Wang et al., 2019). For example, Tobin et al. (2010) prefers "populated places" than to "facilities" such as farms and mines, when there are multiple candidate geospatial labels.

Words in the nearby context are used to disambiguate ambiguous place names (Lieberman and Samet, 2012; WISTR, Speriosu and Baldridge, 2013; Zhang and Gelernter, 2014; Topocluster, DeLozier et al., 2015; CBH, SHS, Kamalloo and Rafiei, 2018; DM_NLP, Wang et al., 2019CamCoder, Gritta et al., 2018; Cardoso et al., 2019; MLG, Kulkarni et al., 2020). Ways of using these context words range from simple to complex. For example, *WISTR* (Speriosu and Baldridge, 2013) uses a simple context window of 20 words on each side of the target place name, and thereby benefits from location-oriented words such as uptown and beach. In contrast, Zhang and Gelernter (2014) searches for common country and state names in other nearby location expressions, in essence, resolving these mostly unambiguous place names to help resolve the target place name.

One sense per referent is a heuristic that assumes that all occurrences of a unique place name in the same document will refer to the same geographical database entry (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; GeoTxt, Karimzadeh et al., 2013; CBH, SHS, Kamalloo and Rafiei, 2018 DM_NLP, Wang et al., 2019). For ex-

ample, after each time that Lieberman et al. (2010) resolves a place name to a geospatial label, it propagates the same resolution to all identical place names in the remainder of the document.

Spatial minimality is a heuristic that assumes that place names in a text tend to refer to geospatial regions that are in close spatial proximity to each other (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; CLAVIN, Berico Technologies, 2012; SPIDER, Speriosu and Baldridge, 2013; Topocluster, DeLozier et al., 2015; CBH, SHS, Kamalloo and Rafiei, 2018;). For example, when Lieberman et al. (2010) geocodes the text *96 miles south of Phoenix, Arizona, just outside of Tucson*, it takes *Tucson* as an "anchor" toponym and resolves that first to get a target region. Then for *Phoenix*, it selects the geospatial label that is most geographically proximate to the target region.

6.3 Implementation Types

Rule-based systems use hand-crafted rules and heuristics to predict a geospatial label for a place name (Edinburgh Parser, Grover et al., 2010; Tobin et al., 2010; Lieberman et al., 2010; Lieberman and Samet, 2011; CLAVIN, Berico Technologies, 2012; GeoTxt, Karimzadeh et al., 2013; Laparra and Bethard, 2020). The rule bases range in size from 2 to more than 200 rules, and rules may be formalized in rule grammars or defined more informally and provided as code. For example, Lieberman et al. (2010) uses a rule (defined via code) to identify place names in comma groups, such as groups of prominent places (e.g., "New York, Chicago and Los Angeles", all major cities in the U.S.), and then resolves all toponyms in the group by applying a heuristic uniformly across the entire group. As another example, Laparra and Bethard (2020) use 219 synchronous grammar rules to parse a target polygon from reference polygons by constructing a tree of geometrical operators (e.g., $BETWEEN(p_1, p_2)$ calculates the region between geolocation polygons p_1 and p_2).

Feature-based machine-learning systems use many of the same features and heuristics of rule-based systems, but provide these as input to a supervised classifier which then makes the prediction of a geospatial label (Martins et al., 2010; MG, Freire et al., 2011; Lieberman and Samet, 2012; WISTR, Speriosu and Baldridge, 2013; Zhang and Gelernter, 2014; Topocluster, DeLozier et al., 2015; CBH, SHS, Kamalloo and Rafiei, 2018; DM_NLP, Wang et al., 2019). They typically operate in a two-step rank-then-rerank framework, where first an information retrieval system produces candidate geospatial labels, then a supervised machine-learning model produces a probability or distance for each candidate, and the candidates are reranked by these scores. Common classification algorithms include logistic regression (WISTR, Speriosu and Baldridge, 2013), support vector machines (Martins et al., 2010; Zhang and Gelernter, 2014), random forests (MG, Freire et al., 2011; Lieberman and Samet, 2012), and stacked LightGBMs (DM_NLP, Wang et al., 2019). For example, Martins et al. (2010) train a support vector machine regression model using features such as the population and the number of alternative names for candidates.

Deep learning systems have approached toponym resolution as a one-step classification problem, by dividing the Earth's surface into a $N \times N$ grid, where the neural network attempts to map place names and their features to one of these $N \times N$ categories (CamCoder, Gritta et al., 2018; Cardoso et al., 2019; MLG, Kulkarni et al., 2020). Each system has a unique neural architecture for combining inputs to make predictions, typically based on either convolutional neural networks (CNNs) or recurrent neural networks (RNNs).

CamCoder (Gritta et al., 2018) was the first deep learning based-geocoder, and combines both lexical and geospatial database features. Its lexical model separately encodes three kinds of input with CNNs: context words (a window of 200 words, location mentions excluded), location mentions (context words excluded) and the target place name. Its geospatial model produces a vector representation using a geospatial label's population (from the database) as its prior probability. CamCoder concatenates the output of the lexical and geospatial models for the final classification. This architecture is illustrated in fig. 3.

MLG (Kulkarni et al., 2020), is also a CNN-based geocoder, but it does not use population or other geospatial database information, and is thus database independent. It captures lexical features in a similar manner to CamCoder, but takes advantage of the S2 geometry[6] to represent its geospatial output space in hierarchical grid-cells from coarse to fine-grained. MLG can predict the geospatial

[6] https://s2geometry.io/

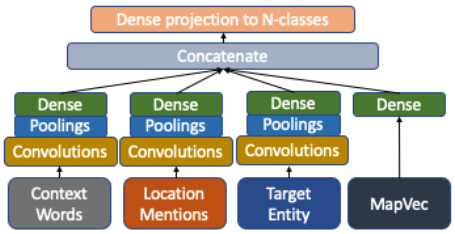

Figure 3: The architecture of CamCoder. The lexical model learns representations for the context words, location mentions, and the target place name, while the geospatial model (called MapVec) draws population information from the geospatial database.

label of a place name at multiple S2 levels by mutually maximizing both precision and generalization of predictions.

Cardoso et al. (2019) proposed a RNN-based database-independent geocoder that uses Hierarchical Equal Area isoLatitude Pixelisation (HEALPix)(Gorski et al., 2005) instead of S2 geometry to discretize the Earth's surface. It considers three inputs: the place name, local context (50 words around the place name), and larger context (paragraph or 500 words around the place name). These are encoded using a Long Short-Term Memory (LSTM) network with Elmo embeddings(Peters et al., 2018), a kind of pre-trained contextual word embedding. The three embeddings for the three inputs are concatenated and used for two types of prediction: the class of HEALPix region, and the coordinates of the centroid of the HEALPix class. This joint learning approach allows the two tasks to be mutually promoted and restricted.

7 Future Directions

A key direction of future research will be output representations. Many past geocoders focused on mapping place names to geospatial database entries (see column 4 of table 2). This was convenient, enabling fast resolution by applying standard information retrieval models to propose candidate entries from the database, but was limited by the simple types of matching that information retrieval systems could perform. Modern deep learning approaches to geocoding allow more complex matching of place names to geospatial locations, but typically rely on discretizing the Earth's surface into tiles to constrain the size of the network's output space. For the neural networks to achieve the fine-grained level of geocoding available in geocoding databases, they may need to consider hierarchical output spaces (e.g., Kulkarni et al. (2020)) or compositional output spaces (e.g., Laparra and Bethard (2020)) that can express the necessary level of detail without exploding the output space.

Another key direction of future research will be the structure and evaluation of geocoding datasets. Most existing datasets and systems treat geocoding as a problem of identifying points rather than polygons (see column 4 of table 1 and column 5 of table 2). Yet the vast majority of real places in geospatial databases are complex polygons (as in fig. 2), not simple points. More polygon-based datasets are needed, especially ones like GeoCoDe (Laparra and Bethard, 2020) that include complex descriptions of locations (e.g., *between the towns of Adrano and S. Maria di Licodia*) and not just explicit place names (e.g., *Paris*). The current state-of-the-art for complex geographical description geocoding is rule-based, but more polygon-based datasets will drive algorithmic research that can improve upon these rule-based systems with some of the insights gained from deep neural network approaches to explicit place name geocoding.

Finally, geocoding evaluation is still an open research area. Future research will likely extend some of the new polygon-based evaluation metrics. For example, using polygon precision and recall would give credit to a geocoding system that predicted the GeoNames entry *Nakhon Sawan* even if the annotated data used the entry *Changwat Nakhon Sawan*, since the polygons of these two place names are nearly identical.

8 Conclusion

After surveying a decade of work on geocoding, we have identifed several trends. First, combining contextual features with geospatial database information makes geocoders more powerful. Second, like much of NLP, geocoders have moved from rule-based systems to feature-based machine-learning systems to deep-learning systems. Third, the older rank-then-rerank approaches, combining information retrieval and supervised classification, are being replaced by direct classification approaches, where the Earth's surface is discretized into many small tiles. Finally, the field of geocoding is just beginning to look beyond a point-based view of locations to a more realistic polygon-based view.

References

Einat Amitay, Nadav Har'El, Ron Sivan, and Aya Soffer. 2004. Web-a-where: geotagging web content. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 273–280.

Zahra Ashktorab, Christopher Brown, Manojit Nandi, and Aron Culotta. 2014. Tweedr: Mining twitter to inform disaster response. In *ISCRAM*, pages 269–272.

Berico Technologies. 2012. Cartographic location and vicinity indexer (clavin).

Preeti Bhargava, Nemanja Spasojevic, and Guoning Hu. 2017. Lithium NLP: A system for rich information extraction from noisy user generated text on social media. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 131–139, Copenhagen, Denmark. Association for Computational Linguistics.

Jens A de Bruijn, Hans de Moel, Brenden Jongman, Jurjen Wagemaker, and Jeroen CJH Aerts. 2018. Taggs: grouping tweets to improve global geoparsing for disaster response. *Journal of Geovisualization and Spatial Analysis*, 2(1):2.

Ana Bárbara Cardoso, Bruno Martins, and Jacinto Estima. 2019. Using recurrent neural networks for toponym resolution in text. In *EPIA Conference on Artificial Intelligence*, pages 769–780. Springer.

Zhiyuan Cheng, James Caverlee, and Kyumin Lee. 2010. You are where you tweet: a content-based approach to geo-locating twitter users. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 759–768.

Grant DeLozier, Jason Baldridge, and Loretta London. 2015. Gazetteer-independent toponym resolution using geographic word profiles. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2382–2388. AAAI Press.

Grant DeLozier, Ben Wing, Jason Baldridge, and Scott Nesbit. 2016. Creating a novel geolocation corpus from historical texts. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 188–198, Berlin, Germany. Association for Computational Linguistics.

Nuno Freire, José Borbinha, Pável Calado, and Bruno Martins. 2011. A metadata geoparsing system for place name recognition and resolution in metadata records. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, pages 339–348.

Krzysztof M Gorski, Eric Hivon, Anthony J Banday, Benjamin D Wandelt, Frode K Hansen, Mstvos Reinecke, and Matthia Bartelmann. 2005. Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759.

Milan Gritta, Mohammad Pilehvar, N Limsopatham, and Nigel Collier. 2017. What's missing in geographical parsing?

Milan Gritta, Mohammad Taher Pilehvar, and Nigel Collier. 2018. Which Melbourne? augmenting geocoding with maps. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1285–1296, Melbourne, Australia. Association for Computational Linguistics.

Milan Gritta, Mohammad Taher Pilehvar, and Nigel Collier. 2019. A pragmatic guide to geoparsing evaluation. *Language Resources and Evaluation*, pages 1–30.

Claire Grover, Richard Tobin, Kate Byrne, Matthew Woollard, James Reid, Stuart Dunn, and Julian Ball. 2010. Use of the edinburgh geoparser for georeferencing digitized historical collections. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 368(1925):3875–3889.

Simon I Hay, Katherine E Battle, David M Pigott, David L Smith, Catherine L Moyes, Samir Bhatt, John S Brownstein, Nigel Collier, Monica F Myers, Dylan B George, et al. 2013. Global mapping of infectious disease. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 368(1614):20120250.

David Jurgens, Tyler Finethy, James McCorriston, Yi Xu, and Derek Ruths. 2015. Geolocation prediction in twitter using social networks: A critical analysis and review of current practice. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 9.

Ehsan Kamalloo and Davood Rafiei. 2018. A coherent unsupervised model for toponym resolution. In *Proceedings of the 2018 World Wide Web Conference*, pages 1287–1296.

Morteza Karimzadeh, Wenyi Huang, Siddhartha Banerjee, Jan Oliver Wallgrün, Frank Hardisty, Scott Pezanowski, Prasenjit Mitra, and Alan M MacEachren. 2013. Geotxt: a web api to leverage place references in text. In *Proceedings of the 7th workshop on geographic information retrieval*, pages 72–73.

Sayali Kulkarni, Shailee Jain, Mohammad Javad Hosseini, Jason Baldridge, Eugene Ie, and Li Zhang. 2020. Spatial language representation with multilevel geocoding. *arXiv preprint arXiv:2008.09236*.

Egoitz Laparra and Steven Bethard. 2020. A dataset and evaluation framework for complex geographical description parsing. In *Proceedings of the 28th International Conference on Computational Linguistics*,

pages 936–948, Barcelona, Spain (Online). International Committee on Computational Linguistics.

JL Leidner. 2007. *Toponym resolution: A comparison and taxonomy of heuristics and methods*. Ph.D. thesis, PhD Thesis, University of Edinburgh.

Michael D Lieberman and Hanan Samet. 2011. Multifaceted toponym recognition for streaming news. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 843–852.

Michael D Lieberman and Hanan Samet. 2012. Adaptive context features for toponym resolution in streaming news. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 731–740.

Michael D Lieberman, Hanan Samet, and Jagan Sankaranarayanan. 2010. Geotagging with local lexicons to build indexes for textually-specified spatial data. In *2010 IEEE 26th international conference on data engineering (ICDE 2010)*, pages 201–212. IEEE.

Inderjeet Mani, Christy Doran, Dave Harris, Janet Hitzeman, Rob Quimby, Justin Richer, Ben Wellner, Scott Mardis, and Seamus Clancy. 2010. Spatialml: annotation scheme, resources, and evaluation. *Language Resources and Evaluation*, 44(3):263–280.

Bruno Martins, Ivo Anastácio, and Pável Calado. 2010. A machine learning approach for resolving place references in text. In *Geospatial thinking*, pages 221–236. Springer.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Michael Speriosu and Jason Baldridge. 2013. Text-driven toponym resolution using indirect supervision. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1466–1476.

Laura Tateosian, Rachael Guenter, Yi-Peng Yang, and Jean Ristaino. 2017. Tracking 19th century late blight from archival documents using text analytics and geoparsing. In *Free and open source software for geospatial (FOSS4G) conference proceedings*, volume 17, page 17.

Richard Tobin, Claire Grover, Kate Byrne, James Reid, and Jo Walsh. 2010. Evaluation of georeferencing. In *proceedings of the 6th workshop on geographic information retrieval*, pages 1–8.

Jan Oliver Wallgrün, Morteza Karimzadeh, Alan M MacEachren, and Scott Pezanowski. 2018. Geocorpora: building a corpus to test and train microblog geoparsers. *International Journal of Geographical Information Science*, 32(1):1–29.

Xiaobin Wang, Chunping Ma, Huafei Zheng, Chu Liu, Pengjun Xie, Linlin Li, and Luo Si. 2019. Dm_nlp at semeval-2018 task 12: A pipeline system for toponym resolution. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 917–923.

Davy Weissenbacher, Arjun Magge, Karen O'Connor, Matthew Scotch, and Graciela Gonzalez-Hernandez. 2019. SemEval-2019 task 12: Toponym resolution in scientific papers. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 907–916, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Wei Zhang and Judith Gelernter. 2014. Geocoding location expressions in twitter messages: A preference learning method. *Journal of Spatial Information Science*, 2014(9):37–70.